A

5-azaCytosine

Vani Brahmachari and Shruti Jain
Dr. B. R. Ambedkar Center for Biomedical Research, University of Delhi, Delhi, India

Synonyms

Azacitidine

Definition

It is the non-methylable analogue of cytosine base. It is not a naturally occurring base but can be incorporated into DNA during replication and into RNA during transcription by growing the cells in media containing 5-azacytidine as a drug or incorporated during in vitro reaction. It is also known to inhibit DNA methyltransferases (DNMTs) thereby, causing lack of methylation in DNA sequence, affecting the interaction between regulatory protein and the nucleic acid target. The inhibition of methylation occurs through the formation of stable complexes between the molecule and DNMTs, thereby saturating cell methylation machinery. It is known to target the CpG islands in the human genome, especially in the promoter regions of genes susceptible to aberrant hypermethylation. Such analogues can be used for modulation of DNA methylation (Kaminskas et al. 2005).

Cross-References

► Epigenetics, Drug Discovery

References

Kaminskas E, Farrell AT, Wang YC, Sridhara R, Pazdur R (2005) FDA drug approval summary: azacitidine (5-azacytidine, Vidaza) for injectable suspension. Oncologist 10(3):176–182

7-Aminoactinomycin D

► Lymphocyte Labeling, Cell Division Investigation

ABCB1

► ATP-binding Cassette B1 Transporter

Abduction

Angelika Kimmig
Departement Computerwetenschappen, Katholieke Universiteit Leuven, Heverlee, Belgium

Definition

Abduction is the task of finding an explanation for an observation with respect to the given knowledge.

W. Dubitzky et al. (eds.), *Encyclopedia of Systems Biology*, DOI 10.1007/978-1-4419-9863-7,

Characteristics

The key principle underlying abduction is that of hypothesizing a set of facts that, together with the available knowledge, explain a specific observation. For instance, given background knowledge about flying and non-flying objects, including a rule stating that normal birds fly $(\forall x.bird(x) \wedge normal(x) \rightarrow flies(x))$, the observation that Tweety flies ($flies(tweety)$) can be explained by abducing that Tweety is a normal bird ($\{bird(tweety), normal(tweety)\}$).

More formally, given background knowledge B, a set A of abducibles (i.e., facts that can be part of explanations), and a ground fact or observation o, the task of abduction is to find an explanation for o, that is, a set of facts $E \subseteq A$ such that o can be inferred from $B \cup E$ using ▶ deduction.

Similarly to ▶ induction, abduction can be seen as a form of inverted ▶ deduction. However, whereas induction finds general rules from several examples, abduction assumes these rules to be part of the background knowledge and instead finds ground facts explaining a single example (Flach and Kakas 2000). The principle of abduction goes back to the philosopher and logician Charles Sanders Peirce, who viewed it as the part of scientific methodology seeking explanations that turn surprising observations into plausible consequences of an existing theory.

In the context of Bayesian networks, finding the most probable explanation (MPE), that is, the most likely values of all non-observed variables given observed values for a subset of variables, is sometimes also considered abductive inference. A similar idea, albeit in the context of a first order logic language and with a clear distinction between abducibles and the rest of the theory, is used in probabilistic horn abduction (PHA) (Poole 1993). PHA associates probabilities to abducibles, thus allowing one to choose explanations based on their probabilities. In abductive logic programming (Kakas et al. 1992), integrity constraints are used to restrict the set of possible explanations.

Cross-References

▶ Deduction
▶ Induction

References

Flach PA, Kakas AC (2000) Abduction and induction – essays on their relation and integration. Kluwer, Dordrecht
Kakas AC, Kowalski RA, Toni F (1992) Abductive logic programming. J Log Comput 2(6):719–770
Poole D (1993) Probabilistic Horn abduction and Bayesian networks. Artif Intell 64:81–129

Abortive Transcription

Nobuo Shimamoto
Faculty of Life Sciences, Kyoto Sangyo University, Kyoto, Japan

Definition

The iterative synthesis of short oligo RNA of typically 2–15 bases long by a promoter complex.

Cross-References

▶ Transcription in Bacteria

Absolute Protein Quantification

▶ Selective Reaction Monitoring

Absorption Spectroscopy

Xiaohua Wu
Department of Pediatrics, Herman B Wells Center for Pediatric Research, Indiana University School of Medicine, Indianapolis, IN, USA

Definition

A technique in which the power of a beam of light measured before and after interaction with a sample is compared.

Cross-References

▶ Spectroscopy and Spectromicroscopy

Abstraction

C. Maria Keet
KRDB Research Centre, Free University of Bozen-Bolzano, Bolzano, Italy

Synonyms

Aggregation; Generalization; Grouping

Definition

Abstraction is used in two principal distinct senses:

1. The process to go from instances to a universal or concept and, in turn, its meta-level
2. The process to go from a detailed to simplified representation where one chooses to ignore certain aspects, thereby reducing the scope toward salient semantics.

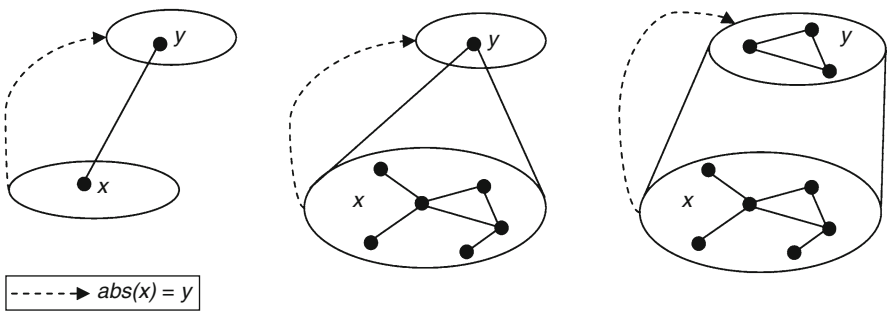
For systems biology, both senses of abstraction are used, where the second sense comes into play only if the former becomes too wieldy at either the instance or the type level. Software support is often required in order to deal with abstractions in systems biology, therefore much effort has gone into research of abstraction. The second sense enjoys attention in particular in conceptual modeling for database systems (Talheim 2009) and artificial intelligence (Giunchiglia et al. 1997) (▶ Knowledge Representation) with as scope to simplify a logical theory of a domain of interest or its visualization, using heuristics, syntactic, or semantic abstraction, whereas the first approach to abstraction is also used to make distinctions between the implementation layer, logical design layer, and conceptual modeling layer.

Characteristics

Abstraction Mechanisms

There are at least three principle mechanisms of abstraction in the second sense, which are graphically depicted in Fig. 1, and each mechanism can have variations. For instance, for R-ABS, one can take the abstraction from the constituent parts (▶ Mereology) or members to the whole (e.g., the individual sheep into a flock, enzymatic reaction into a metabolic pathway process) and for F-ABS, one "folds" (e.g., customer, payment, and book onto a "BookOrder" entity or the steps in the MAPK cascade).

Implicitly, abstraction forces into existence a notion of *levels* or *layers*, and successive abstractions then generate an *abstraction hierarchy*, which is a close relative of levels of ▶ granularity and relates to the notion of ▶ modularity and ▶ modularization of a system. Not all approaches toward, and usages of, abstraction consider levels and hierarchies explicitly and reify them into entities themselves that can be manipulated. For those that do, there are different



Abstraction, Fig. 1 Three conceptually distinct types of abstraction operations; *x* and *y* are entities at a finer and coarser level of abstraction (indicated with an *oval*), and *x* is abstracted into *y* using an abstraction function *abs*. *R-ABS* the relation is remodeled as a function, *F-ABS* folding multiple entities and relations into a different type of entity, *D-ABS* deleting semantically less relevant entities and relations (Source: based on Keet (2007))

ways to formalize abstraction and its supporting levels and hierarchies. In the first sense of abstraction, it is common for the contents at each (explicit or implicit) abstraction level to have its own language and vocabulary to represent the data, information, or knowledge at that level. In the second sense of abstraction, the proposed formalizations aim to solve this within one logic language.

Limitations

To date, there is no overwhelming evidence that the solutions proposed by computing are implementable either in information systems for systems biology or modeling for systems biology. Where it already can be useful is to provide guidelines to create abstraction levels in a consistent way, that is, not ad hoc but with guiding principles and abstraction procedures.

Cross-References

► Granularity

References

Giunchiglia F, Villafiorita A, Walsh T (1997) Theories of abstraction. AI Commun 10(3–4):167–176

Keet CM (2007) Enhancing comprehension of ontologies and conceptual models through abstractions. In: Basili R, Pazienza MT (eds) 10th congress of the Italian association for artificial intelligence (AI*IA 2007), Springer Lecture Notes in Artificial Intelligence 4733, pp 814–822

Talheim B (2009) Abstraction. In: Liu L, Tamer Ozsu M (eds) Encyclopedia of database system. Springer, New York

AC#

► Database Accession Number

Activating Complexes

► Trithorax Complexes

Activation Domain Shielding

Tetsuro Kokubo
Department of Supramolecular Biology, Graduate School of Nanobioscience, Yokohama City University, Yokohama, Kanagawa, Japan

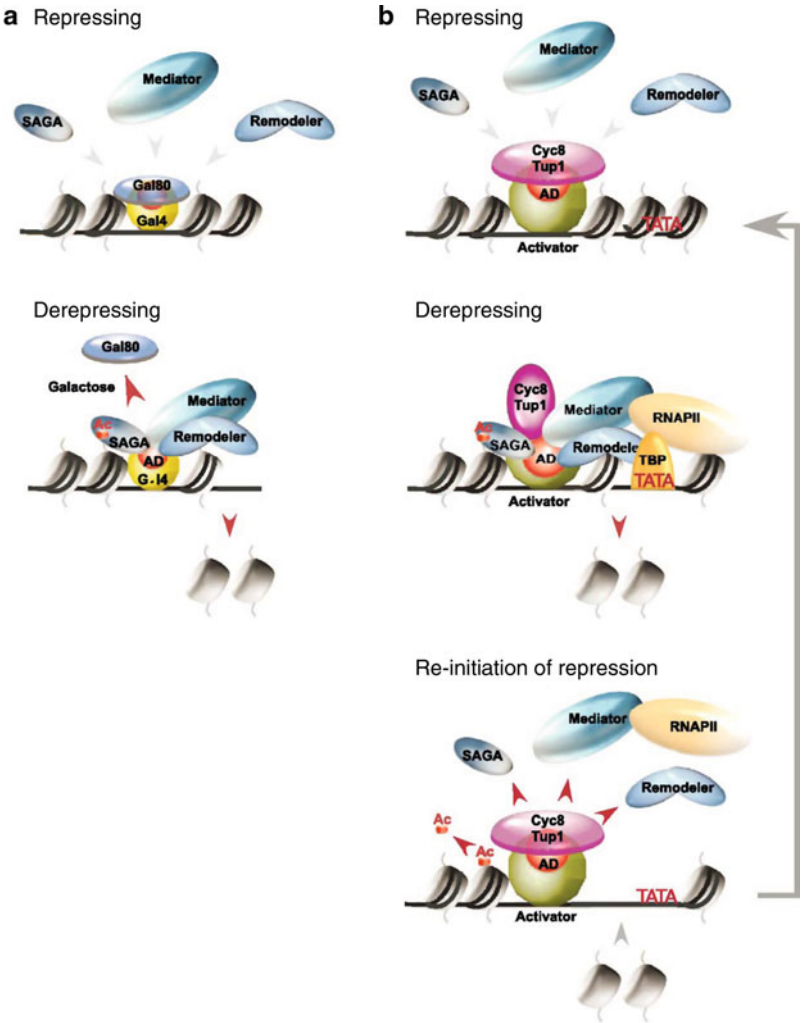
Synonyms

Shielding of activation domain

Definition

In budding yeast, several *GAL* genes including *GAL1*, *GAL10*, and *GAL7* are activated by Gal4p, which is a transcriptional activator for the *GAL* regulon. Expression of the *GAL* genes allows the cell to utilize the sugar galactose as a carbon source. In the absence of galactose, Gal4p is bound to DNA but is inactive because its activation domain (AD) is masked by Gal80p (Fig. 1a) (Wong and Struhl 2011). Once galactose is added to the media, the signal transduction protein Gal3p binds the sugar along with ATP to allow activation of Gal4p. Specifically, the galactose- and ATP-bound form of Gal3p can form a stable complex with Gal80p, thereby titrating out the inhibitory effect of Gal80p on Gal4p (Lavy et al. 2012). It is not known whether Gal80p is released from the transcription complex upon activation or instead remains in complex with Gal3p, possibly in a different position or orientation. Regardless, upon binding of Gal3p and Gal80p, the AD of Gal4p becomes able to recruit a range of transcriptional coactivators. Such shielding of ADs may represent the simplest mode of action by which transcriptional corepressors like Gal80p negatively regulate gene expression.

The Tup1p-Cyc8p complex of budding yeast is a well-known transcriptional corepressor that represses transcription of several hundreds of genes. This complex is widely conserved among eukaryotes and is therefore likely to be crucial for proper regulation of gene expression. Previously, it was believed that Tup1p-Cyc8p represses transcription by several



Activation Domain Shielding,
Fig. 1 Transcriptional repression by shielding the activation domains (ADs) of transcription activators. (**a**) Regulation of Gal4p by Gal80p. Under repressing conditions, the AD of Gal4p is shielded by Gal80p. In derepressing conditions, Gal80p dissociates from Gal4p, exposing the AD of Gal4p for recruitment of various coactivators. (**b**) Regulation of some activators by Tup1p-Cyc8p. Under repressing conditions, Tup1p-Cyc8p prevents activation by shielding the ADs of these activators. Upon derepression, these activators may undergo specific conformational changes that affect their interactions with Tup1p-Cyc8p, such that the ADs can now recruit various coactivators. In contrast to Gal80p, Tup1p-Cyc8p remains at the promoter, where it contributes somehow to the activation process. When repression needs to be reestablished, Tup1p-Cyc8p may help to remove these coactivators from the promoter

distinct mechanisms that function redundantly, e.g., by recruiting histone deacetylases (HDACs), positioning nucleosomes at inhibitory sites, or inhibiting Mediator function at promoters. Recently, however, it has been proposed that the major function of Tup1p-Cyc8p is to shield ADs from interacting with coactivators such as SWI/SNF, SAGA, and Mediator (Fig. 1b) (Wong and Struhl 2011). In this regard, Tup1p-Cyc8p and Gal80p are functionally similar, although the former is not specific for the *GAL* regulon and functions more widely than the latter. Consistent with this newer model, the binding sites of Tup1p-Cyc1p are located very close to those of coactivators, as revealed by studies of Tup1p-depleted cells. Therefore, at least some transcriptional repressors can be converted to transcriptional activators by exchange of their corepressor accessory factors for coactivating factors. However, it remains possible that some other transcriptional repressors may repress transcription more actively, e.g., in the manner previously proposed for Tup1p-Cyc8p.

Cross-References

▶ Mechanisms of Transcriptional Activation and Repression

References

Lavy T, Kumar PR, He H, Joshua-Tor L (2012) The Gal3p transducer of the GAL regulon interacts with the Gal80p repressor in its ligand-induced closed conformation. Genes Dev 26(3):294–303

Wong KH, Struhl K (2011) The Cyc8-Tup1 complex inhibits transcription primarily by masking the activation domain of the recruiting protein. Genes Dev 25(23):2525–2539

Activator

Jianhua Ruan
Department of Computer Science, University of Texas at San Antonio, San Antonio, TX, USA

Definition

A DNA-binding protein that stimulates transcription of one or more genes. Most activators recruit RNA polymerase to promoter region by interacting directly with a subunit of DNA polymerase and serving as a liaison between RNA polymerase and DNA, or change DNA conformation to promote the transition to the open complex required for initiation of transcription.

Active Labeling During Cell Division

▶ Quantifying Lymphocyte Division, Methods

Active Labeling Incorporation

▶ Lymphocyte Labeling, Cell Division Investigation

Active Learning

Jan Ramon
Declarative Languages and Artificial Intelligence Group, Katholieke Universiteit Leuven, Leuven, Belgium

Synonyms

Experimental design; Learning by queries

Definition

Given a set of unlabeled training examples, an active learning algorithm is allowed to select a number of training examples (one by one or in batches) and to obtain their target value at a certain cost per example. Next, the learning algorithm should predict the target value of a number of unseen test examples, incurring a certain cost per mistake. The goal of the active learning algorithm is to minimize the total cost.

Characteristics

Obtaining target values of training examples may be expensive. For example, in experimental research requiring microarrays, bioassays, patients, clinical trials, mass-spectrography, crystallography, or other physical ▶ experiments in order to collect data, the amount of available resources limits the amount of experimental data which can be obtained.

In an active learning setting, the learning algorithm must take this economic reality into account. Every target value has a cost, and once a model is learned every prediction mistake also has a cost. The goal is to be as economical as possible.

Problem Settings

There are several variations of the active learning problem. First, active learning can be seen as learning by asking queries to some oracle (the teacher, experimental setup, etc.) (Angluin 1988). Different query types have been defined, among which the most important ones are:

- The *membership query*, which asks whether a particular example belongs to a particular class or not
- The *equivalence query*, which asks whether a particular learned model is equivalent to the concept to be learned and asks for a counterexample if the learned concept is not correct

The equivalence query is very powerful and therefore not very useful in practical applications. In particular, it is usually not possible to verify whether a theory is perfect (due to the absence of perfect experimentation equipment, resources or sufficiently knowledgeable experts), and for many datasets it is even unlikely that a perfect theory can be derived. The membership query lets the oracle return the target value of a given example, and is more usual in most experimental

research. In the case of noisy experiments, the membership query may only give an approximative value and it may be useful to ask the same query again to get a more precise measurement.

Second, variation is possible in the interaction procedure between the learner and the oracle. Most common is the situation where the learner asks queries one by one and the oracle provides the answers immediately. In practice, this is not always realistic. State-of-the-art high-performance experimental setups (microarrays, bioassays, ...) process several experiments simultaneously in batches or in pipelines. It is therefore more efficient in practice to ask a next query before the first one has been answered.

Third, different prediction objectives are possible. In the normal setting of active learning, the algorithm is scored on prediction performance on a test set drawn from the same distribution as the training set (Kearns and Vazzirani 1994). However, one is not always most interested in learning a model of all aspects of the instance space. In some cases, one wants to find the best instance in the instance space according to some criterion. Accordingly, one is interested in a model that is especially accurate for the instances that are good while an accurate prediction of the value of bad instances is not very important. For example, in drug discovery (De Grave et al. 2008), one wants to find the "best" (most active, least toxic, ...) molecule in a library of molecules.

Finally, one can distinguish between membership query–based approaches where the learning algorithm should provide the instances for which the class should be obtained, and the membership query–based approaches where the algorithm can choose from a predefined pool of instances. The latter is more realistic, as in the first case it may happen that the learner asks to perform a practically hard or impossible to realize experiment. For example, for drug discovery processes, large libraries of purchasable compounds are available. Other molecules can be tested too, but having to synthesize such molecules oneself drastically increases the cost of the experiment.

Applications

Active learning can be applied in a wide range of different applications.

In principle, it is most useful in experimental research. Only few systems have been implemented in a completely automated closed loop where the output of the active learning algorithm is used directly by an experimental setup. One successful application was in the field of functional genomics (King et al. 2004).

Even so, the value of active learning strategies has been demonstrated in several application areas, either on benchmark data or in comparative experiments with standard experimental designs. These domains include drug discovery (De Grave et al. 2008), nanofiltration (Cano-Odena et al. 2010), sensor placement (Guestrin et al. 2005), and robot gait optimization (Lizotte et al. 2007).

Methods

Several methods for active learning have been described in the literature, for example, (Angluin 1988; Tong and Koller 2001a; Lizotte et al. 2007; Guestrin et al. 2005). Here, we mainly summarize a few general principles:

- When learning a predictive model, a common strategy is to select instances for which the prediction using the current model is most uncertain. This is a particularly useful strategy for predictive models which also report confidence, such as Gaussian processes (Guestrin et al. 2005).
- Alternatively, for version space–based algorithms, a good strategy is to try as well as possible to halve the version space with each query. For example, when the parameters of a model are known to be in some region, it is good to choose instances to query such that this region is maximally reduced by the answer. Tong and Koller (2001b) discuss the application of such strategies for support vector machines.
- When choosing several instances to query, a trade-off must be made between informativeness and diversity of the instances. In particular, choosing several informative but very similar instances may not give a maximal total amount of new information. For example, in drug discovery one often investigates molecules similar to molecules known to have some activity in the hope to find a better variant, but one also performs a wider screening in the hope to find new regions in molecular space where active compounds can be found.
- Another idea arises from ▶ ensemble learning. When several different models are learned on the same data, they may agree on a part of the test instances and disagree on another part of the test instances. In such a case, Liere and Tadepalli (1997) argues that it is good to query the target values of

the instances on which the committee of classifiers learned so far disagrees most.
- When attempting to find the best instances in a set, one will focus on the best instances according to the quality criterion. A typical strategy among several alternatives (De Grave et al. 2008) is to select instances optimistically, for example, by adding to the current prediction a few standard deviations on that prediction. This makes a trade-off between selecting instances which will probably have a high quality and selecting instances in poorly explored regions of the instance space.

Cross-References

- ▶ Ensemble
- ▶ Experiment

References

Angluin D (1988) Queries and concept-learning. Mach Learn 2:319–342

Cano-Odena A, Spilliers M, Dedroog T, De Grave K, Ramon J, Vankelecom IFJ (2010) Micropollutant removal via genetic algorithms and high throughput experimentation. J Membr Sci 366(12):25–32

De Grave K, Ramon J, De Raedt L (2008) Active learning for high throughput screening. In: Proceedings of the eleventh international conference on discovery science, Budapest. Lecture Notes in Computer Science, vol 5255, pp 185–196

Guestrin C, Krause A, Singh AP (2005) Near-optimal sensor placement in Gaussian processes. In: Proceedings of the 22nd international conference on machine learning, Bonn, pp 265–272

Kearns M, Vazzirani U (1994) An introduction to computational learning theory. MIT Press, Cambridge, MA

King RD, Whelan KE, Jones FM, Reiser PG, Bryant CH, Muggleton SH, Kell DB, Oliver SG (2004) Functional genomic hypothesis generation and experimentation by a robot scientist. Nature 427:247–252

Liere R, Tadepalli P (1997) Active learning with committees for text categorization. In: Proceedings of the 14th conference of the American association for artificial intelligence (AAAI-97), Providence, pp 591–596

Lizotte D, Wang T, Bowling M, Schuurmans D (2007) Automatic gait optimization with Gaussian process regression. In: Proceedings of the 20th international joint conference on artificial intelligence, Hyderabad, pp 944–949

Tong S, Koller D (2001) Active learning for structure in Bayesian networks. In: Proceedings of the seventeenth International joint conference on artificial intelligence (IJCAI), Seattle. Morgan Kaufman, Washington, pp 863–869

Tong S, Koller D (2001b) Support vector machine active learning with applications to text classification. J Mach Learn Res 2:45–66

Actomyosin Ring

Sebastian Mana-Capelli and Dannel McCollum
Department of Molecular Genetics and Microbiology, University of Massachusetts, Worcester, MA, USA

Synonyms

Contractile actomyosin ring (CAR); Contractile ring

Definition

The actomyosin ring is the most archetypical structure of cytokinesis. The ring contains actin filaments cross-linked by the motor protein type II MYOSIN as well as numerous other actin cytoskeletal components. Constriction of the actomyosin ring results in furrow ingression and cytokinesis. The precise mechanisms that connect the contractile ring to the cell membrane are still not clear (Eggert et al. 2006). The ingression force generated by this structure has been traditionally explained through the "purse and string" model (Schroeder 1972) whereby bipolar myosin filaments walk along actin filaments to bring about constriction of the ring in a manner similar to muscle constriction. As constriction proceeds, acting filaments are disassembled and as a result the thickness of the contractile ring remains approximately constant.

Cross-References

- ▶ Cytokinesis

References

Eggert US, Mitchinson TJ, Field CM (2006) Animal cytokinesis: from parts list to mechanisms. Annu Rev Biochem 75:543–566

Schroeder TE (1972) The contractile ring: II. Determining its brief existence, volumetric changes, and vital role in cleaving *Arbacia* eggs. J Cell Biol 53:419–434

Acute Upper Respiratory Tract Infections

▶ Viral Respiratory Tract Infections

Acyclic Digraph

▶ Directed Acyclic Graph

Adaptation

Philippe Huneman
Institut d'Histoire et de Philosophie (IHPST), des Sciences et des Techniques, Université Paris 1 Panthéon-Sorbonne, Paris, France

Definition

"Adaptation" may name a process or a state, can be used in physiological or evolutionary contexts, and concerns organisms or traits. An individual organism has abilities to physiologically adapt to its environment, for example, by changing the values of some parameters of its metabolism (pulse, body temperature, etc.). Its being adapted is the result of such physiological process. In evolutionary biology, it may be useful to talk of *adaptedness* of organisms, namely, their being adjusted to their environments, and of *traits* themselves as *adaptations*. Adaptedness is always relative to an environment. Traits as adaptations, fitness and adaptedness of organisms are basically related by natural selection; namely, the process by which the *more adapted* organisms, having higher chances of survival and reproduction (i.e., higher *fitness*), on the average leave more offspring and hence increase the frequency of their heritable traits in the next generations and finally lead to the fixation of those traits, namely, the *adaptations*.

In neo-Darwinian biology, a trait is an adaptation when it results from natural selection (Burian 2005). To this extent, adaptation is defined by natural selection; calling a trait an adaptation is therefore a historical statement. The longstanding amazement of naturalists in front of the adaptation of species to their milieu has been explained by Darwin's hypothesis of natural selection as the main process accounting for organismal traits. Natural selection explains the presence of a trait, but this explanation is not complete if one does not know an "adaptation for what" the trait is, i.e., to which selective pressures it owes its existence (Brandon 1990). Notably, a trait can be shown to be an adaptation, i.e., shaped by natural selection, whereas we do not know an adaptation for what (e.g., Kreitmann tests on genome sequences may show they result from selection but give no idea of the reasons for which they have been selected).

Yet in the context of the explanation of maintenance of traits, some behavioral ecologists defined an adaptation as the fittest phenotypic variant, notwithstanding its evolutionary origin (Reeve and Sherman 1993), especially because many evolutionary explanations do not take the historical context into account.

Concerning the link between adaptations and adaptedness, one could argue that the more adaptations an organism has, the more adapted it is; in this sense, evolutionary biologist Julian Huxley called organisms "bundles of adaptations" (though this not exactly being his own view), yet this position can be criticized as too much adaptationist. The set of adaptations characterizing organisms constitutes what is often called its "design." Given that natural selection somehow increases inclusive fitness, it is plausible to say that organisms are designed to maximize their inclusive fitness.

However, two adaptations for distinct environmental demands can be conflicting, and therefore they do not increase the fitness of organisms in an additive manner. For example, risky behavior in some species are shown to have the function of attracting female mates (e.g., through the "handicap principle," see ▶ Sexual Selection), yet it conflicts with many adaptations which increase the survival chances of the organisms (fear reactions, etc.)

With a given genotype, some individuals may display a range of various phenotypes adapted to their environment – this is called "phenotypic plasticity." For instance, some species of fish will turn into either males or females depending on the temperature of water. It is useful to distinguish this ▶ plasticity from the evolutionary notion of adaptation, which refers to the genotypic underpinning of traits.

Cross-References

- ▶ Cell Cycle Checkpoints
- ▶ Explanation, Evolutionary

References

Brandon R (1990) Adaptation and environment. MIT Press, Cambridge
Burian R (2005 [1983]) Adaptation. In: Burian R (ed) The epistemology of development, evolution, and genetics. Cambridge University Press, Cambridge, pp 54–80
Reeve HK, Sherman P (1993) Adaptation and the goals of evolutionary research. Quart Rev Biol 68:1–32

Adaptationism

Philippe Huneman
Institut d'Histoire et de Philosophie (IHPST), des Sciences et des Techniques, Université Paris 1 Panthéon-Sorbonne, Paris, France

Definition

A position which assumes that each trait of an organism is the result of an optimization process, driven by natural selection, and then is somehow optimal. Steven Jay Gould and Richard Lewontin (1978) famously forged the term and criticized it because it overlooks the fact that organisms display a cohesive unity and because adaptationist explanations are often not falsifiable. However, Godfrey-Smith (2001) usefully distinguished three brands of adaptationism: adaptationism can be either *methodological* (Maynard-Smith 1984) and then unlikely to be right or wrong; or an *empirical* claim, whose testing is not yet uncontroversial; or finally *explanatory*, namely, expresses the belief that the most "interesting traits" (e.g., complex adaptations) are due to selection (yet the notion of "interesting" is irreducibly subjective). Methodological adaptationism emphasizes that optimality models allow one to discover *constraints* – e.g., developmental or historical constraints – which prevent real organisms to reach the optima predicted by the models; therefore, it does not underlie an empirical claim that organic nature is perfectly adapted or that natural selection always leads to optimization.

Cross-References

- ▶ Explanation, Evolutionary

References

Godfrey-Smith P (2001) Three kinds of adaptationism. In: Orzack SH, Sober E (eds) Adaptationism and optimality. Cambridge University Press, Cambridge, pp 335–357
Gould SJ, Lewontin R (1978) The spandrels of san marco and the panglossian paradigm: a critique of the adaptationist programme. Proc R Soc Lond B 205:581–598
Maynard-Smith J (1984) Optimization theory in evolution. Ann Rev Ecol Syst 9:31–56

Adaptive Immune Response Cascade

- ▶ Adaptive Immune System

Adaptive Immune System

Shoba Ranganathan
Department of Chemistry and Biomolecular Sciences and ARC Center of Excellence in Bioinformatics, Macquarie University, Sydney, NSW, Australia
Department of Biochemistry, Yong Loo Lin School of Medicine, National University of Singapore, Singapore

Synonyms

Adaptive immune response cascade; Adaptive immunity

Definition

The adaptive immune system is a collective term given to a group of highly specialized, systematic cells and processes that prevent vertebrates from certain death by pathogenic infections (Alberts et al. 2002).

Cross-References

- ▶ Major Histocompatibility Complex (MHC)
- ▶ TR Recognition of MHC-Peptide Complexes

References

Alberts B, Johnson A, Lewis J, Raff M, Roberts K, Walter P (2002) The adaptive immune system. In: Molecular biology of the cell, 4th edn. Garland Science, New York, pp 1363–1421

Adaptive Immunity

- ▶ Adaptive Immune System

Ada-Two-A-Containing

- ▶ ATAC

Adhesin

Ramachandran Srinivasan
G.N. Ramachandran Knowledge Centre for Genome Informatics, Institute of Genomics and Integrative Biology, Delhi, India

Synonyms

Adhesion molecule

Definition

Adhesins are cell surface–localized proteins helping in the process of adhesion. In pathogens, adhesins constitute major virulence factor helping in the process of adherence and colonization.

Adhesion Molecule

- ▶ Adhesin

Adipocytes

Steven D. Rhodes
School of Medicine, Indiana University, Indianapolis, IN, USA

Definition

Adipocytes are responsible for the storage of energy in the form of lipids (fat). Adipocytes are derived from mesenchymal stem cells and reside in the adipose tissue. Adipocyte differentiation can be induced in vitro from mesenchymal stem cell cultures in the presence of factors such as dexamethasone, isobutilmethylxanthine, indomethacin, and insulin. Adipocytes can be identified phenotypically by positive Oil red O staining, indicating the accumulation of lipid droplets within the cytoplasm. Molecular markers for adipocytes include peroxisome proliferator-activated receptor gamma 2 (PPARγ2), C/EBPβ, aP2, adipsin, leptin, and lipoprotein lipase.

Cross-References

- ▶ Single Cell Assay, Mesenchymal Stem Cells

Adjuvants

Gajendra Raghava
Bioinformatics Centre, Institute of Microbial Technology, Chandigarh, Chandigarh, India

Definition

These are molecules used to improve the efficacy of vaccine mostly through delivery of the vaccine in controlled release mode and at the desired location.

Adult T-Cell Leukemia

Christian Schönbach
Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, Iizuka, Fukuoka, Japan

Synonyms

Adult T-cell leukemia/lymphoma; ATL; ATLL

Definition

HTLV-1 is the only known human retrovirus with oncogenic potential. About 5% of infected HTLV-1 individuals develop ATL, a non-Hodgkin type lymphoma with abnormal mature $CD4^+$ T-cell phenotypes and extremely long latency of up to 60 years. HTLV-1 infections and Tax-mediated transformation of $CD4^+$ T-cells alone is not sufficient to trigger ATL. Other factors that are necessary to develop ATL include proviral load, progressive weakening of the host immune system, genetic background, virus-mediated genetic and epigenetic changes resulting in DNA damage, and clonal expansion of altered $CD4^+$ T-cells (Higuchi and Fujii 2009).

Infections with HTLV-2 may result in HAM/TSP-like disease, but do not cause ATL or other forms of leukemia/lymphoma. Although HTLV-2 is transforming infected $CD4^+$ T-cells, differences in Tax proteins of HTLV-1 and HTLV-2 and their molecular interactions with host proteins are thought to be responsible for their distinct disease outcomes (Higuchi and Fujii 2009).

Typical ATL symptoms are skin lesions, lymphadenopathy, and hepatosplenomegaly. ATL is often accompanied by hypercalcemia and increased osteoclast numbers. Clinically, ATL is divided into four subclasses (Table 1) (Takatsuki 2005).

At present, there is no effective treatment of ATL. Chemotherapy, INFA2 plus antiviral treatment, humanized monoclonal IL2 antibody therapy, or allogeneic hematopoietic stem cell transplantation yield mixed results (Takatsuki 2005). Nevertheless, INFA2 combined with antiviral zidovudine (AZT) treatment prolonged the survival time and improved clinical symptoms in patients with acute, chronic, and smoldering ATL, except lymphoma-type ATL (Bazarbachi et al. 2010).

Adult T-Cell Leukemia, Table 1 Clinical subclasses of ATL

ATL subclasses	Symptoms
Acute	High number of ATL cells; frequent skin lesions, systemic lymphadenopathy, and hepatosplenomegaly
Chronic	Slightly elevated white blood cell count; skin lesions, lymphadenopathy, and/or hepatosplenomegaly
Smoldering	Low number ATL cells; skin or pulmonary lesion
Lymphoma	Systemic lymphadenopathy; low number of abnormal cells in peripheral blood

Cross-References

► HTLV, Cellular Transcription

References

Bazarbachi A, Plumelle Y, Ramos JC, Tortevoye P, Otrock Z, Taylor G, Gessain A, Harrington W, Panelatti G, Hermine O (2010) Meta-analysis on the use of Zidovudine and Interferon-Alfa in adult T-Cell leukemia/lymphoma showing improved survival in the leukemic subtypes. J Clin Oncol 28(27):4177–4183. Epub ahead of print

Higuchi M, Fujii M (2009) Distinct functions of HTLV-1 Tax1 from HTLV-2 Tax2 contribute key roles to viral pathogenesis. Retrovirology 6:117

Takatsuki K (2005) Discovery of adult T-cell leukemia. Retrovirology 2:16

Adult T-Cell Leukemia/Lymphoma

► Adult T-Cell Leukemia

Adverse Events

Ravi Iyengar
Department of Pharmacology and Systems Therapeutics, Mount Sinai School of Medicine, New York, NY, USA

Definition

Complications and/or undesirable side effects that are associated with pharmacological treatments, which can be the result of unpredicted interactions.

Cross-References

- ▶ Biomarker Discovery, Knowledge Base
- ▶ Systems Pharmacology

Aerobic Glycolysis

- ▶ Warburg Effect

Agent-based Modeling

Zhihui Wang and Thomas S. Deisboeck
Harvard-MIT (HST) Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Charlestown, MA, USA

Synonyms

Individual-based modeling (IBM)

Definition

Agent-based modeling (ABM), also referred to as individual-based modeling (IBM), simulates the interactions of autonomous entities (i.e., the agents) with each other and their local environment to predict higher-level emergent phenomena (Bonabeau 2002). In biomedical research, while an agent can represent a part of a cell or a cluster of cells, the ideal candidate for a software agent is now more commonly recognized to be an individual cell (Walker and Southgate 2009), since models can benefit from such a direct one-to-one mapping between real and virtual cells in terms of parameter acquisition from experiments and model validation. As a simulation progresses, agents (representing individual cells) interact or communicate with other agents and their common microenvironment according to a set of predefined, biomedical data-driven "rules." Because an ABM's simulation results are highly dependent on these rules, it is necessary to tightly couple these algorithms at all stages of model development with iterative in silico studies as well as available in vitro or in vivo biological experiments in order to validate and calibrate these rules according to relevant data.

In the field of cancer research, the use of ABM to simulate cancer growth per se is not new (Wang and Deisboeck 2008). Some sophisticated ABMs have been developed to identify and quantify the relationship between individual molecular properties, their microenvironmental conditions, and the overall tumor morphology. It should be noted that current ABMs often model extracellular factors as continuous quantities, thereby rendering the models hybrid in nature. The ABM approach allows researchers to investigate one of the most important problems in cancer research – tumor heterogeneity, an inherent feature of cancer cells. This is achieved by addressing the role of diversity in cell populations and also within each individual cell. In current ABMs, tumor growth and invasion patterns are neither predefined nor intuitive, but emerge as a result of individual dynamics, which include cell–cell and cell–microenvironment interactions and intercellular signaling. However, a major drawback of ABMs is that they are generally too detailed to simulate over a long period of time, particularly in large, 3D domains, and as a consequence current ABMs can only process a relatively small number of cells. Hence, a more innovative way of thinking about ABM is required to solve this problem in order to move ABMs toward clinical application. We have seen a number of new methods being developed in the field (interested readers should refer to Wang and Deisboeck (2008) for details).

Cross-References

- ▶ Multilevel Modeling, Cell Proliferation

References

Bonabeau E (2002) Agent-based modeling: methods and techniques for simulating human systems. Proc Natl Acad Sci USA 99(Suppl 3):7280–7287
Walker DC, Southgate J (2009) The virtual cell–a candidate coordinator for "middle-out" modelling of biological systems. Brief Bioinform 10:450–461
Wang Z, Deisboeck TS (2008) Computational modeling of brain tumors: discrete, continuum or hybrid? Sci Model Simul 15:381–393

Agent-based Models, Discrete Models and Mathematics

Shayn M. Peirce
Department of Biomedical Engineering, University of Virginia, Charlottesville, VA, USA

Synonyms

Individual-based modeling; Multi-agent simulation

Definition

Agent-based modeling (ABM) is a computational modeling approach whereby individual entities in a complex system (e.g., biological cells) are represented by discrete agents that interact autonomously in simulated space and time to produce emergent, often nonintuitive outcomes at the population (e.g., biological tissue) level. ABM has roots in the field of artificial intelligence and can be considered an offshoot of cellular automata (CA) modeling (► Cellular Automata), which was first introduced in the 1940s by John Von Neumann (von Neumann and Morgenstern 2007). Unlike CA, however, agents in an ABM are not confined to stationary points on a grid (i.e., they are mobile), they do not have to perform their actions simultaneously at each time step, and they can be heterogeneous in size or type (Grimm and Railsback 2005). With the advent of personal computers in the 1980s, the adoption of ABM accelerated and expanded into different fields, including ecology, epidemiology, finance and economics, political science, the social and behavioral sciences, and more recently, biological patterning. As applied to biological and biomedical investigation, ABMs are particularly well suited for investigating multicellular phenomena, or processes where the independent behaviors of individual cells acting in response to their dynamic local environment give rise to emergent tissue-level adaptations. In this context, ABMs have been used to simulate and study a variety of physiological and pathological processes including tumorigenesis, inflammation, wound healing, and angiogenesis (Peirce et al. 2006). The rules governing individual agent behaviors are central to the outcomes/predictions of ABMs and often incorporate the stochasticity inherent in biology. Outputs of ABM are frequently graphical in nature and easily characterized by patterning metrics that define the architectural features of cellular organizations at the tissue level. Together, these attributes make ABM a useful tool in biological and biomedical research for investigating how complex and even somewhat random interactions among individual cells at lower levels of spatial scale integrate to produce emergent results at higher levels of scale.

Characteristics

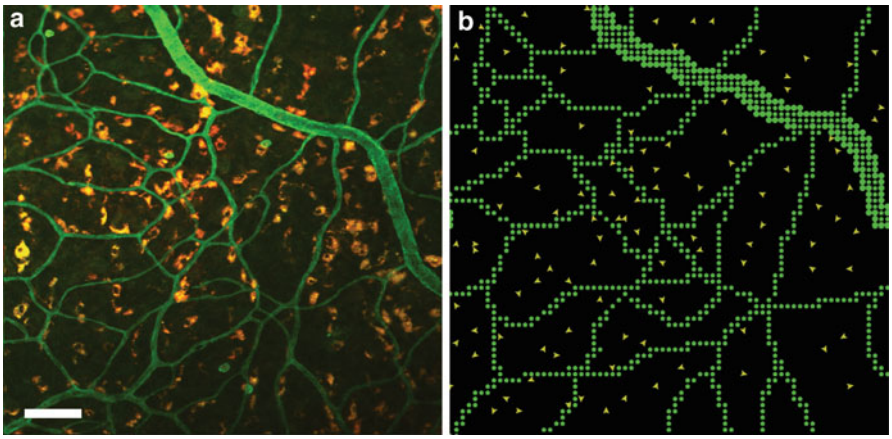
Assumptions

There are a number of underlying assumptions that are typically made when constructing an ABM to study biological tissue patterning, and these are summarized below:

- Agents are finite in number, and unless otherwise specified, each agent represents a single biological cell within a tissue.
- Biological cells are treated as "black boxes" in that biological phenomena at the subcellular level (e.g., gene expression or intracellular signaling) are implicit and reflected in the behaviors and attributes of agents at the cell level.
- Agents operate within a bounded space whose boarders are analogous to those of the biological tissue being simulated.
- Space and/or time is represented discretely. A typical two-dimensional ABM simulation space, for example, is partitioned into a grid of pixels on which agents act and interact with one another and with their simulated environment.
- The state of knowledge in a given field can be effectively captured by the rule set that governs agent behaviors in an ABM.
- If the prediction of the ABM matches independently measured outcomes of a biological analogue, the rule set may, but not necessarily, capture key mechanisms controlling the biological system.

Formulation

Historically, biological ABMs have been used for two primary purposes: (1) to replicate the emergent behaviors of a complex biological system in order to better understand its underlying mechanisms and (2) to address specific fundamental questions/hypotheses



Agent-based Models, Discrete Models and Mathematics, Fig. 1 The multicell composition of a complex tissue (rat mesentery) includes microvascular endothelial cells (*green*) arranged in a network of vessels and interstitial progenitor cells (*yellow*) visualized using immunohistochemistry and confocal microscopy (**a**) (▶ Fluorescence Microscopy). An ABM represents cells as agents and can simulate patterning outcomes at the level of the microvascular network (**b**)

about the underlying mechanisms of complex biological systems. The process of formulating an ABM is iterative, and few, if any, formalized protocols for the construction of biological ABMs exist; however, a general recommended stepwise approach for developing an ABM is summarized below.

First, the simulation space and the agents are defined according to the tissue being simulated and its cellular and acellular composition (Fig. 1). Specifically, the simulation boundaries are set by the size of the tissue being simulated (in two dimensions or three dimensions). Spatial boundaries, or edges of the simulation space, can be periodic, or wrapping from left to right and top to bottom, in the case of a two-dimensional simulation. An alternative boundary condition is a closed boundary, wherein the edges of the simulation space are rigid and serve as "walls" to prevent agents from exiting the space. The number and phenotype of cells/agents contained within the tissue and the extracellular composition of the tissue are defined and ascribed to agent-types in the ABM.

Second, the agents are endowed with the ability to exhibit the relevant physiological or pathological behaviors. For example, cellular agents can proliferate, migrate, undergo apoptosis, differentiate, and/or secrete proteins, all in response to other signals in their simulated environment. Likewise, acellular agents (e.g., collagen) may be endowed with the ability to remodel in response to enzymes in the environment (e.g., matrix metalloproteinase) or bind and release growth factors.

A reasonable third step is the selection of initial conditions, an appropriate time step (clock increments during which events in the simulation are scheduled), and a total duration (i.e., temporal window) for the simulation. Frequently, time steps are discrete and represent from millisecond to hours. During each time step, the agents survey the environment, make decisions based on their local environment that are dictated by the rule set, and exhibit behaviors. ABMs can simulate temporal windows ranging from minutes to years.

The fourth step involves determining the relevant output metrics that capture the salient features of the emergent outcome at the tissue level. It is often useful to design metrics that quantitatively reflect the shape or architecture of the tissue at different time steps, and/or patterns of cell aggregates and multicell structures within the tissue. An obvious guideline for selecting metrics is to include those that are utilized to describe analogous patterning features in experimental data, as this facilitates cross-validation of the ABM predictions against independent empirical studies.

The fifth step involves compiling the set of rules that determine the behaviors of the individual agents in the ABM. Rules can be constructed as abstract representations of known biological relationships or as specific representations of actual empirical data. Frequently rules within a rule set are crafted as Boolean, true-false "if-then" statements, but they can also incorporate

stochasticity and describe agent behaviors according to probability distributions. An example of a rule that might exist in a two-dimensional ABM simulation meant to represent cell contact inhibition would be: "If an agent is within one pixel of a neighboring agent for more than 24 h, that agent will not divide." The length of a biological ABM rule set can range from one to over 200 rules, and can be limited by computational power and the available empirical data.

The first five steps described above pertain to the design and assembly of the model, and the sixth step is run and verify model by determining whether or not the output is internally consistent and, at least at an intuitive level, aligned with representative biological outcomes. This step can also include performing additional simulations in order to conduct a sensitivity analysis on the parameter/rule set and to understand if the unperturbed system reaches dynamic equilibrium or steady state after multiple time steps.

The final step in developing and using an ABM to study biological processes involves performing an independent validation of the predictions by comparing ABM outcomes to experimentally measured outcomes and using the model to interrogate the underlying mechanisms of the complex system. This may include performing simulations where agents or rules are systematically removed or altered to understand the relative contributions of those factors to the overall system. Removing the influences of key growth factors or chemokines one by one, for example, would enable a "knockout" analysis to elucidate how the system performs in the absence of these factors. One might perform such an analysis in order to independently predict phenotypes observed in an analogous transgenic knockout mouse. This step might also include testing alternative hypotheses by implementing them as rules in the ABM rule set and comparing the different predictions to empirically measured results. This type of analysis might suggest a subset of plausible working hypotheses to pursue experimentally, thus providing a tool to inform the design of experiments in the wet lab and streamlining and/or accelerating the overall discovery process (Hunt et al. 2009). A common misconception about ABMs is that they are only informative if they recapitulate features of the biological system. On the contrary, ABMs can be highly instructive when they fail to accurately predict measured biological outcomes because they can point to specific voids, inconstancies, and/or errors in data or understanding and suggest specific experiments that would offer the most value moving forward.

In summary, developing and using ABMs is an iterative process that is most informative when married with experimental studies to determine the underlying cell-level mechanisms of tissue patterning (Thorne et al. 2007). By integrating different types of experimental data into a unified rule set and allowing agent/cells to reveal the outcome of this integration in space and time, ABMs can uncover new understanding that is not obtainable using experimental approaches alone. ABMs may play a role in translational research, informing drug design and target identification, as well as predicting side effects and determining mechanisms of action (Vodovotz et al. 2010). Ultimately, ABMs should be considered as another tool in the repertoire of systems biology approaches that is particularly useful in studying complex, multicell processes and outcomes at the tissue level.

Strengths and Weaknesses of ABM

Strengths

- Multicell behaviors can be represented at the level of the tissue and individual cells can be tracked in space and time in a relatively computationally efficient manner.
- Heterogeneities in cell distribution and tissue composition, as well as stochastic changes in biological events, can be modeled explicitly.
- ABMs produce graphical outputs that are easily interpreted by nonexperts in computational modeling, enabling a potentially rich and fluid dialogue between experimentalists and modelers.
- ABMs integrate diverse types of experimental data and serve to instantiate existing and new hypotheses to elucidate mechanistic relationships across complex biological systems.

Weaknesses

- Because ABMs can be very sensitive to small variations in the rule set and rarely account for physical conservation laws explicitly, their predictions can be highly irregular or nonbiological. Therefore, the risk of trivial or irrelevant predictions should be mitigated by rigorous independent validation against experimental data.
- ABMs that incorporate stochastic rules require multiple runs to fully explore the extent of possible outcomes.

- Depending on the size and scope of the model, ABMs can have large numbers of rules and parameters, which may make parameter identification difficult and require extensive sensitivity analyses to determine the robustness of the predictions.
- ABMs do not explicitly account for intracellular events and serve as abstractions of the biology at this level, thus limiting their ability to provide meaningful results regarding gene regulation, metabolic processes, and intracellular signaling, for example.

ABM Tools

There are a number of different ABM software platforms that have been developed by individuals and consortiums. NetLogo, MASON, Repast, and SWARM have general applicability across a range of disciplines, including biology, and have some of the more active user communities. Each of these software platforms has distinguishing features that make them more appealing than others for certain applications, as reviewed previously (Railsback and Lytinen 2006). Other ABM software platforms have been developed for specific fields of study (e.g., finance, education, social sciences).

Cross-References

- ▶ Cellular Automata
- ▶ Fluorescence Microscopy

References

Grimm V, Railsback SF (2005) Individual-based modeling and ecology, Princeton series in theoretical and computational biology. Princeton Classic Editions, Princeton

Hunt CA, Ropella GE, Lam TN, Tang J, Kim SH, Engelberg JA, Sheikh-Bahaei S (2009) At the biological modeling and simulation frontier. Pharm Res 26:2369–400

Peirce SM, Skalak TC, Papin JA (2006) Coupling intracellular networks with tissue-level physiology. IBM J Res 50:1–15

Railsback SF, Lytinen SL (2006) Agent-based simulation platforms: review and development recommendations. Simulations 82:609–623

Thorne BC, Bailey AM, Peirce SM (2007) Combining experiments with multi-cell agent-based modelling to study biological tissue patterning. Brief Bioinform 8:245–257

Vodovotz Y, Constantine G, Faeder J, Mi Q, Rubin J, Bartels J, Sarkar J, Squires RH Jr, Okonkwo DO, Gerlach J, Zamora R, Luckhart S, Ermentrout B, An G (2010) Translational systems approaches to the biology of inflammation and healing. Immunopharmacol Immunotoxicol 32:181–95

von Neumann J, Morgenstern O (2007) Theory of games and economic behavior (commemorative edition). Princeton Classic Editions, Princeton

A

Agglomerative Hierarchical Clustering

▶ Hierarchical Agglomerative Clustering

Agglomerative Hierarchical Data Segmentation

▶ Hierarchical Agglomerative Clustering

Aggregation

▶ Abstraction

Aggregation Relation

▶ Mereology
▶ Meronymy

Aging

Rajeswara Babu Mythri[1], Shireen Vali[2] and M. M. Srinivas Bharath[1]
[1]Department of Neurochemistry, National Institute of Mental Health and Neurosciences (NIMHANS), Bangalore, Karnataka, India
[2]Cell Works Group Inc., Bangalore, India

Definition

Aging is classically defined as a long-term physiological process associated with morphological and functional changes in the human body at the tissue and cellular level, aggravated by injury resulting in a progressive imbalance of the regulatory systems including neuronal, hormonal, and immune mechanisms. Human life

expectancy has nearly doubled since the twentieth century causing tremendous increase in aged population (65 years and older). The increasing number and severity of health problems associated with aging presents one of the major health care challenges in the world. This includes the increased propensity of aged population to encounter neurological diseases.

Physiological aging is an important causative factor underlying the onset of sporadic PD. This is because the cumulative changes in the midbrain contribute to specific degenerative pathways and PD pathology. For example increased oxidative stress manifested by accumulation of oxidatively modified proteins increases with age. This increase magnifies several folds in late age due to a combination of increased production of reactive species, decreased antioxidant activity, and impaired ability to repair or remove the modified proteins. Additionally, activities of the protein clearing cellular machinery including the ubiquitin-proteasome system (UPS) and autophagy decline with age thereby reducing the neuron's ability to remove damaged or modified proteins or protect itself from damaging free radicals. This also leads to neurotoxic aggregation of uncleared proteins in the cell. Such cellular insults are compounded by other age-associated pathways such as mitochondrial dysfunction, accumulation of iron, increased DA oxidation, increased deposits of lipids, etc. which increase with aging.

Cross-References

► Disease System, Parkinson's Disease

Agonist

Melissa L. Kemp
The Wallace H. Coulter Department of Biomedical Engineering, Georgia Institute of Technology and Emory University, Atlanta, GA, USA

Definition

An agonist is a ligand that binds to a cell surface receptor and triggers a response from the cell. An agonist may be endogenous, i.e., a natural ligand that binds to its receptor under in vivo conditions, or it can be a synthetic, exogenous ligand.

Akaike's Information Criterion (AIC)

Kejia Xu
Institute of Systems Biology, Shanghai University, Shanghai, China

Definition

Akaike's Information Criterion (AIC) is a technique that measures the goodness of an estimated statistical model and selects a model from a set of candidate models. The chosen model is the one that is expected to minimize the difference between the model and the truth. Given a data set, several competing models may be ranked according to their corresponding AIC, and the one having the lowest AIC will be the best.

In the general case, the AIC is defined as

$$AIC = 2k - 2\ln\ L, \tag{1}$$

where k is the number of parameters in the statistical model, and L is the maximized value of the likelihood function for the estimated model.

References

Akaike H. A new look at the statistical model identification. IEEE Trans Autom Contr. 1974;19(6):716–23

Alarmins

► Damage-associated Molecular Patterns

Algorithmic Modeling Languages

► Cell Cycle Modeling, Process Algebra

AliBaba

Conrad Plake[1] and Jörg Hakenberg[2]
[1]Biotechnology Center (BIOTEC), Technische Universität Dresden, Dresden, Germany
[2]Department of Computer Science and Department of Biomedical Informatics, Arizona State University, Tempe, AZ, USA

Definition

AliBaba is a text mining (see ▶ Applied Text Mining) system that analyzes scientific abstracts, extracts biomedical entities such as genes and diseases (▶ Named Entity Recognition), and displays the results as a graph depicting relationships between these entities, such as protein–protein interactions (Plake et al. 2006). AliBaba is available at: alibaba.informatik.hu-berlin.de.

Characteristics

Literature searches form a substantial part of day-to-day work in life science–related domains. Researchers track recent developments in their fields, search for answers to specific questions, or obtain overviews of known facts on a given topic. The PubMed (see ▶ MEDLINE and PubMed) interface to Medline is the most renowned search engine in the life sciences, indexing over 20 million abstracts from more than 5,000 journals. Manually searching this vast amount is a tedious task, especially when exhaustive information are needed, or when information about a set of objects are required. AliBaba helps users in analyzing large PubMed search results by automatically extracting facts that are important in molecular biology and molecular medicine, and by arranging them in an intuitive graphical representation. The software extracts both objects and their interrelationships, forming edges and nodes of the graph, respectively. Users can also manually edit graphs (adding, changing, or deleting nodes and edges) to prepare them for further usage or screenshots. An interface to KEGG allows to overlay extracted networks with known pathways, which enables users to quickly compare KEGG pathways to recent findings in the literature.

Usage

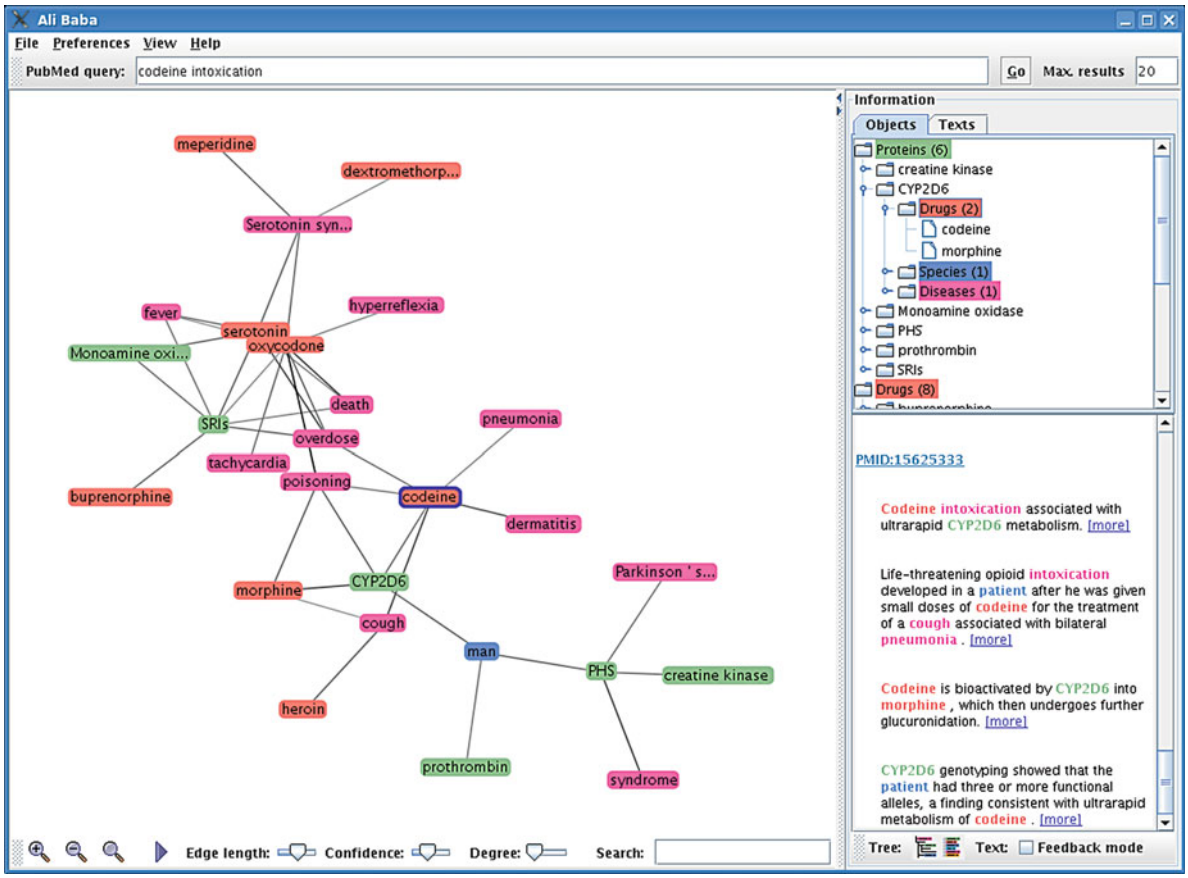
AliBaba builds on the Java Web Start technology and launches a client on the user's machine. The client accepts a PubMed query, sends it to the AliBaba server who in turn retrieves and analyzes the results from PubMed and gets back annotated abstracts. The query syntax and semantics are exactly the same as already known to PubMed users, and AliBaba also keeps the order of results. The results are parsed and shown on a graphical user interface mainly consisting of three panels that show different aspects of the gathered information (see Fig. 1).

Use Case

A patient with cough received standard treatment with codeine and fell unresponsive after a while. To search a reason, "codeine intoxication" is a meaningful PubMed query, but it results in 162 abstracts. Posing this query to AliBaba immediately shows a path from codeine through poisoning to morphine and further to CYP2D6 (Fig. 1). This indicates the solution for this case, where the patient was suffering from a CYP2D6 mutation leading to the ultrarapid metabolism of codeine to morphine.

Named Entity Recognition

Recognition of named entities (names referring to proteins, species, drugs, diseases, etc.) is based on pre-compiled word lists. Recognition of these names is performed using class-specific fuzzy searches. Slight spelling variations (plural, capitalization, hyphenation) are allowed for all classes. For species, we expand the word lists with common types of abbreviations (where they are not already contained in the database), such as "*S. cerevisiae*" for the entry "*Saccharomyces cerevisiae*." Protein names appear with different variations, such as Arabic to Roman numbering conversions ("factor 7" versus "factor VII"). AliBaba maps all recognized entities to their respective entries in the source databases. This enables the user to quickly access further information, for instance, protein sequences or functional descriptions not contained in the current set of PubMed abstracts. In addition, this mapping helps to overcome the problem of synonymity. This arises when



AliBaba, Fig. 1 A drug-disease-protein network extracted from a PubMed search result and shown in AliBaba. The network shows a connection between codeine (marked in the graph with a *blue frame*), cough, poisoning, and CYP2D6. Poisoning is likewise connected to morphine and CYP2D6. The reason is that codeine is bioactivated by CYP2D6 into morphine, which can lead to life-threatening intoxication in certain patients, who show an ultrarapid form of this metabolism mark

different authors use different names for the same entity (for example, CD95, Fas, and Apo-1 for the same protein).

Relation Mining

AliBaba follows two different strategies for finding associations between entities. The first is based on the occurrence of two entities in the same sentence. In this case, AliBaba calculates the confidence score for each pair of entities depending on the number of entities that occur between them. Other studies have shown that the precision of co-occurrence-based methods can be as high as 94% for gene-disease and other associations; but it is, for example, less than 50% for protein–protein interactions, thus requiring different approaches. AliBaba's second strategy is based on aligning sentences against a predefined set of patterns learned from a training corpus. This technique is used to find protein–protein interactions and cellular locations of proteins. For protein–protein interactions, the method achieves a precision of 79% and a recall of 52%, as evaluated on an independent corpus. An external evaluation on the ▶ BioCreative II.5 and the FEBS Letters Experiment on Structured Digital Abstracts II IPS test corpus showed a recall of 69%. The alignment score also defines the quality of a potential match which is reflected in the confidence score.

Related Tools

Tools related to AliBaba are EBIMed (▶ Retrieving and Extracting Entity Relations from EBIMed) and iHOP.

Cross-References

- ▶ Applied Text Mining
- ▶ BioCreative II.5 and the FEBS Letters Experiment on Structured Digital Abstracts
- ▶ MEDLINE and PubMed
- ▶ Named Entity Recognition
- ▶ Retrieving and Extracting Entity Relations from EBIMed

References

Plake C, Schiemann T, Pankalla M, Hakenberg J, Leser U (2006) AliBaba: PubMed as a graph. Bioinformatics 22(19):2444–2445, doi: 10.1093/bioinformatics/btl408. http://dx.doi.org/10.1093/bioinformatics/btl408

Alignment, Protein Interaction Networks

▶ Graph Alignment, Protein Interaction Networks

Allergen

Ramachandran Srinivasan
G.N. Ramachandran Knowledge Centre for Genome Informatics, Institute of Genomics and Integrative Biology, Delhi, India

Definition

Allergens are substances (proteins, carbohydrates, particles, pollen grains, etc.) to which the body mounts a hypersensitive immune response typically of Type I.

α-Helix, β-Sheet, Loop Carbon-Alpha Positioning

▶ Protein Structure Metapredictors

Alternative Routes

Ernesto Perez-Rueda
Departamento de Ingenieŕıa Celular y Biocatálisis, Instituto de Biotecnologia, Universidad Nacional Autonóma de México, Cuernavaca, Morelos, Mexico

Definition

Two main hypotheses have been proposed to explain how the metabolic pathways actually originated: the ▶ retrograde hypothesis and the ▶ patchwork hypothesis.

Recently, another mechanism associated to the metabolical growth suggests the existence of alternative routes or alternologs, defined as branches or enzymes that, proceeding via different metabolites, converge in a common end product. These alternative branches contribute to genetic buffering similar to gene duplication. It has been proposed that the origin of alternative branches is closely related to different environmental metabolite sources and lifestyles among species.

Cross-References

- ▶ Evolution of Metabolism, Amino Acid Biosynthesis Pathways

References

Conant GC, Wagner A (2003) Asymmetric sequence divergence of duplicate genes. Genome Res 13:2052–2058
Hernandez-Montes G, Diaz-Mejia JJ, Perez-Rueda E, Segovia L (2008) The hidden universal distribution of amino acid biosynthetic networks: a genomic perspective on their origins and evolution. Genome Biol 9:R95
Horowitz NH (1945) On the evolution of biochemical syntheses. Proc Natl Acad Sci USA 31:153–157

Alternative Splicing

Luiz O. F. Penalva
Department of Cellular and Structural Biology, Greehey Children's Cancer Research Institute, University of Texas Health Science Center, San Antonio, TX, USA

Definition

For the majority of genes, RNA transcribed from DNA must be processed to become a mature messenger RNA, which is later translated into protein. One RNA processing step is splicing, which involves the removal of intervening sequences, or introns, and connection of exons, which contain the genetic information required for protein synthesis. Greater than 95% of the annotated multi-exons contain protein-coding genes that are alternatively spliced. Alternative splicing is an important mechanism for gene regulation as well as a mechanism to create a diverse proteome given a limited number of genes encoded by the genome. Biologically, alternative splicing can create different protein products, which may be differentially expressed both temporally and spatially. Splicing is catalyzed by the spliceosome, a large macromolecular complex composed of five main small nuclear ribonucleoproteins (snRNP) in concert with many other auxiliary proteins. However, the decision to remove or include particular exons or splice sites is dictated by two elements, sequence elements encoded by the RNA (which usually reside at the exon-intron junction) and *trans*-acting proteins. Depending on the splicing event desired, the *trans*-acting proteins are subdivided into four categories based on its action: exonic splicing enhancers, exonic splicing silencers, intronic splicing enhancers, and intronic splicing silencers.

Many common alternative splicing events can occur; these include exon skipping, intron retention, and alternative 5' splice site selection, alternative 3' splice site selection, mutually exclusive exons, alternative promoters, and alternative polyadenylation. Exon skipping is by far the most common splicing event. On an average, exons are 50–200 bp in length while intronic sequences are usually thousands of base pairs long.

The role of alternative splicing is important as it is a premier mechanism (>50 % of transcripts) for defining tissue specificity. As such, tissue-specific mechanisms exist to dictate alternative splicing constituting mainly of tissue-specific splicing factors. For example, the brain, where the highest number of alternative splicing occur in the body, express several brain-specific splicing factors such as nPTB, NOVA1, NOVA2, and the Hu/ELAV group of proteins.

Cross-References

▶ Post-transcriptional Regulatory Networks

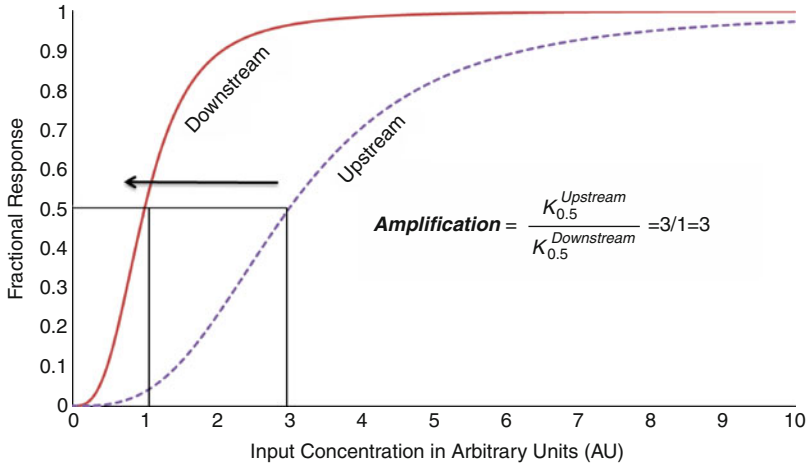
Amplification

Kareenhalli V. Venkatesh
Department of Chemical Engineering, Indian Institute of Technology Bombay, Powai, Mumbai, Maharashtra, India

Definition

Signaling networks contain modules that help the cell in responding to weak extracellular stimulus. Such a capability to respond arises from the ability of the signaling pathway to amplify a weak stimulus and is termed as amplification. There are two kinds of signal amplification in biological systems, namely, magnitude amplification and sensitivity amplification. Magnitude amplification occurs when the output response molecules are produced in larger number than the input signal (stimulus) molecules, while the sensitivity amplification occurs when the percentage change in response is higher than the percentage change in stimulus (Koshland et al. 1982).

Amplification can be characterized by plotting the steady state input-output response for a given input signal for an upstream and downstream effector in a signaling pathway. The shift in the curve of a dose response toward the left, for activation of a downstream component as compared to an upstream component demonstrates amplification (see Fig. 1). Amplification is quantified using the half-saturation constant ($K_{0.5}$) obtained using Hill Equation of the



Amplification,
Fig. 1 Amplification in signaling pathways. *Dotted curve*: Response of the upstream component; *Solid curve*: Response of the downstream component

input-output response curves. The ratio of the half-saturation constants for an upstream component to that of a downstream component to changes in an input signal provides the measure of amplification, and is defined as follows:

$$\text{Amplification} = \frac{K_{0.5}^{\text{Upstream}}}{K_{0.5}^{\text{Downstream}}} \qquad (1)$$

where $K_{0.5}^{\text{Upstream}}$ and $K_{0.5}^{\text{Downstream}}$ are the half-saturation constants of the upstream and downstream components of a signaling pathway obtained using Hill equation (see Fig. 1). Cascade of covalent modification cycles is known to yield amplification. A classic example is the activation of MAPK pathway in Xenopus oocytes. Hundredfold amplification in MAPK and 30-fold amplification in MAPKK have been observed with respect to the upstream kinase, MAPKK (Huang and Ferrell 1996). Signal amplification increases along the cascade and the amplification at particular step depends upon the signal amplitude of the preceding step (Heinrich et al. 2002). The extent of amplification depends on the ratio of the counteracting kinase to phosphatase reaction rates, where the rate of phosphatase should be lesser than that of kinase to bring about signal amplification.

The cascading of signals through multiple steps helps in amplifying a weak signal along with increasing the response sensitivity. The extent of amplification decreases with increasing strength of the signal and vice versa. There is certain threshold for the input stimulus above which the amplification ability of the cascade is lost due to the saturation of the upstream component. If the upstream component is saturated (more than 90%) for a given input value and for the same input value if the downstream component is also saturated, then the role of signal amplification is insignificant (see Fig. 1). Amplification aids the cell to respond to the smallest variations in the stimulus allowing it to respond to a broad range of stimulus strengths. However, the downside of amplification is that the design motif also amplifies the signal noise along with the input signal (Shibata and Fujimoto 2005).

Cross-References

- Hill Equation
- Pathway Modeling, Metabolic
- Signaling Network Resources

References

Heinrich R, Neel BG, Rapoport TA (2002) Mathematical models of protein kinase signal transduction. Mol Cell 9(5):957–970
Huang CYF, Ferrell JE (1996) Ultrasensitivity in the mitogen-activated protein kinase cascade. Proc Natl Acad Sci USA 93(19):10078–10083
Koshland DE, Goldbeter A, Stock JB (1982) Amplification and adaptation in regulatory and sensory systems. Science 217:220–225
Shibata T, Fujimoto K (2005) Noisy signal amplification in ultrasensitive signal transduction. Proc Natl Acad Sci USA 102(2):331–336

Analysis of Variance

Larissa Stanberry
Bioinformatics and High-throughput Analysis Laboratory, Seattle Children's Research Institute, Seattle, WA, USA

Synonyms

ANOVA

Definition

The analysis of variance evaluates the null hypothesis of no difference in means across multiple treatment groups.

Characteristics

Consider an experiment to compare the effectiveness of various feed supplements on the growth rate of chickens. Newly hatched chicks were randomly allocated into six groups, and each group was given a different feed supplement. Their weights in grams after 6 weeks were recorded along with feed types. The number of chickens for each supplement were: `casein:12, horsebean:10, linseed:12, meatmeal:11, soybean:14, sunflower:12`. The data set `chickwts` is freely available as a part of R package. In this example, the response variable is chick weight at 6 weeks and the treatments are different feed supplements.

Figure 1 shows the boxplots for each feed supplement. The weight at 6 weeks appears to vary considerable across the different feed types. Can the variability in the weight be explained by feed type? To answer this question, one could perform all possible pairwise comparisons between the different treatment groups. However, this approach would inflate the experiment-wise error rate. Analysis of variance (ANOVA) tests the hypothesis of no variation due to treatment against the alternative that there exists a pair of treatments for which the mean responses differ.

ANOVA compares the variability across treatments to the variability within treatments. If there is no treatment effect, the two measures should be about the same. If there is a strong treatment effect, one would expect the variability between the treatments to be considerably large than that within the treatment. To test the hypothesis, one can consider the ratio of the between- to within-variance. The ratio is large if there is a strong treatment effect. If the data are random samples from normal distribution and the variance is constant across the treatment groups, under the null hypothesis of no treatment effect, the ratio of the variances follows an F-distribution with parameters given by the degrees of freedom of the numerator and denominator terms. The p-value is obtained by comparing the observed value of the statistic against the F-distribution.

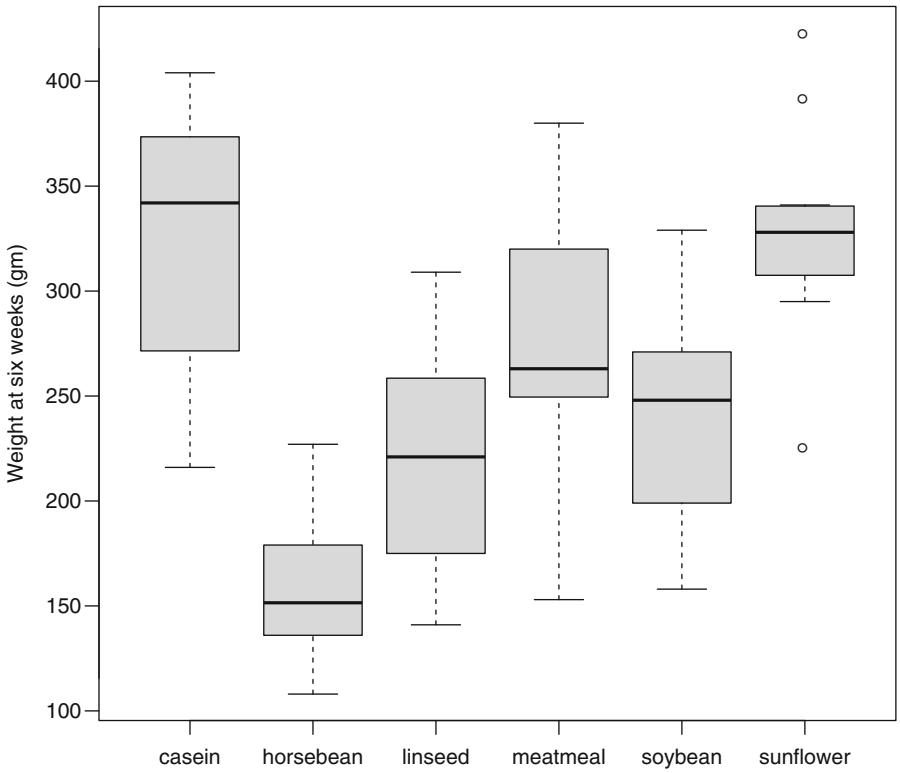
The following ANOVA table gives the summary of the analysis for the chick weight data. This is an example of a one-factor ANOVA. The table gives the degrees of freedom, the sum of squares, and the mean square for the factor and the residuals. The mean square is given by the ratio of the sum of squares over the corresponding degrees of freedom. The mean squares value for "feed" gives an estimate of the variability between the different feed types. The residuals mean squares measure the within-treatment variability.

The F-statistic is computed as a ratio of the mean squares of "feed" to the mean squares of residuals. The p-value can be computed using Permutation Test or Randomization Test. Alternatively, assuming that the data is normally distributed and the error variance is constant, the F-statistic follows an $F(5,65)$-distribution. The large value of the F-statistic indicates a significant effect of the feed type on chick weight at 6 weeks. Note that before interpreting the results, it is important to verify the model assumptions.

```
Analysis of Variance Table
Response: weight
          Df Sum Sq Mean Sq F value    Pr(>F)
feed       5 231129   46226  15.365 5.936e -10 ***

Residuals 65 195556    3009
```

Although ANOVA results indicate the difference in mean weight for different feed types, they provide little guidance as to which treatments exactly differ from each other and in which direction. The investigators are presumably interested in understanding which feed is associated with the largest/smallest weight measurements. To answer this question, we can follow up the ANOVA with a series of t-tests. Multiple comparisons increase the experiment-wise error rate. In this



Analysis of Variance, Fig. 1 The boxplot showing the weight gain separately for the two factors

example, performing all 15 pairwise comparisons with the Type I error of 0.05 would imply the experiment-wise error of approximately 0.75. To control the experiment-wise error rate, one can use the Bonferroni correction method, Fisher's least-significant-difference method, and other suitable procedures.

Cross-References

- ▶ Hypothesis Testing
- ▶ Multiple Hypothesis Testing
- ▶ Permutation Test
- ▶ Randomization Test

References

Montgomery D (2009) Design and analysis of experiments, 7th edn. Wiley, Hoboken, NJ

Analysis of Variance (ANOVA) Tables

- ▶ Experimental Design, Variability

Analysis Situs

- ▶ Topology and Toponomics

Anaphase-Promoting Complex (APC/C)

Sergio Moreno
Instituto de Biología Molecular y Celular del Cáncer, CSIC/Universidad de Salamanca, Salamanca, Spain

Synonyms

Cyclosome

Definition

Multiprotein E3 ligase complex involved in the ubiquitylation of proteins for their degradation by the proteasome. It is activated by Cdc20 or Cdh1, which

specify substrate recognition, promoting anaphase progression, mitotic exit, or the maintenance of G1.

Cross-References

► Anaphase-Promoting Complex Inhibitors
► CDK Inhibitors

Anaphase-Promoting Complex Inhibitors

Masamitsu Sato
Department of Biophysics and Biochemistry, Graduate School of Science, University of Tokyo, Bunkyo–ku, Tokyo, Japan
PRESTO, Japan Science and Technology Agency, Kawaguchi, Saitama, Japan

Synonyms

Anaphase-Promoting Complex (APC/C); Cyclosome

Definition

APC/C (anaphase-promoting complex/cyclosome) is an E3 enzyme (ubiquitin ligase) that degrades substrates including cyclin B and securin, to trigger anaphase onset. An E3 enzyme brings its substrates into close proximity of a ubiquitin conjugating enzyme E2 so that E2 can transfer ubiquitin to the substrate. APC/C is a 1.5 MDa protein complex composed of at least a dozen subunits. Activation of APC/C is essential for anaphase onset during mitosis. In meiosis, in order to ensure two consecutive rounds of chromosome segregation in meiosis I and meiosis II, APC/C activity needs to be partially inhibited during the interkinesis period (the period between meiosis I and II). This is operated by APC/C inhibitors Mes1 in fission yeast and Erp1/Emi2 in *Xenopus* oocytes.

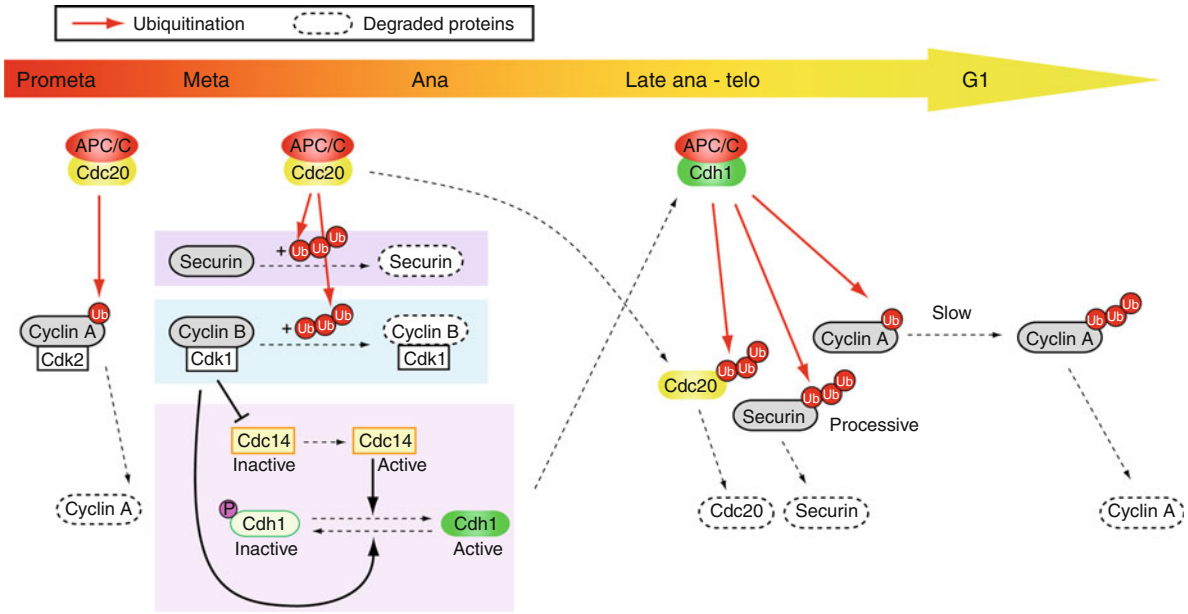
Characteristics

Substrates of APC/C

The substrates of APC/C are key regulators of mitosis, including Aurora kinases and the Polo-like kinase Plk1, securin and cyclin B (reviewed in Peters 2006). Securin is an inhibitor of separase, a protease that cleaves cohesin at anaphase onset. Securin is degraded when all kinetochores are attached by spindle microtubules and therefore the cell is ready to go into anaphase. Destruction of securin in turn activates separase so that it can cleave cohesin. Another substrate cyclin B is required for mitotic/meiotic progression until metaphase, but is downregulated by APC/C at anaphase onset onward. In general, APC/C substrates contain motifs such as the destruction box (D-box) or the KEN box. These motifs are directly recognized by APC/C co-activators Cdc20 and Cdh1 (see below).

Co-activators of APC/C

APC/C is activated by co-activators Cdc20/Slp1/Fizzy and Cdh1/Ste9/Fzr (Fizzy-related). These co-activators contain WD40 repeats as well as the IR-tail at the C-termini. Those proteins are thought to be important for substrate recognition and specificity of APC/C. One of the most important aspects of APC/C function is how ordered destruction of substrates is achieved. As shown in Fig. 1, there are mainly three stages of the cell cycle when APC/C operates: prometa-metaphase, anaphase, and G1 phase. The potential activity of APC/C-Cdc20 increased during prometaphase, but the degradation of other substrates such as cyclin B and securin is delayed by spindle assembly checkpoint (see below). During prometaphase, cyclin A is degraded, whereas securin and cyclin B are degraded at anaphase onset. In late anaphase, "substrate switch" occurs because APC/C degrades its co-activator Cdc20 and replaces it with Cdh1. In budding yeast, Cdh1 is activated by Cdc14, the protein phosphatase that functions antagonistic to Cdk1. APC/C-Cdh1 contributes to the degradation of residual cyclin B and other substrates such as Plk1 and Cdc20. Both Cdc20 and Cdh1 recognize D-boxes, but Cdh1 can also interact with KEN-boxes. This differential recognition system by Cdc20 and Cdh1 should contribute to the ordered destruction of substrates. In addition, the processivity of polyubiquitination is also important for temporal regulation of destruction of Cdh1 substrates (Rape et al. 2006). Among Cdh1 substrates, for instance, securin is shown to be polyubiquitinated by APC/C-Cdh1 in a processive



Anaphase-Promoting Complex Inhibitors, Fig. 1 A timeline of APC/C activation and the substrate specificity. APC/C has many kinds of substrates, but the timing of destruction is temporally organized by co-activators (Cdc20/Slp1/Fizzy and Cdh1/Ste9/Fzr). *Red arrows* indicate ubiquitination (Ub) by APC/C with co-activators

manner, while it takes long time for cyclin A to be polyubiquitinated (Fig. 1).

Inhibition of APC/C

For transition from meiosis I to meiosis II, inhibition of APC/C seems to be a conserved and essential system to ensure the consecutive M phases (see Fig. 2 of the essay ► Meiosis). Moreover, inhibition of premature activation of APC/C before anaphase is also an important aspect of APC/C regulation. There are many kinds of APC/C inhibitors that play roles in M phase progression, as exemplified below.

The Spindle Assembly Checkpoint Components

Mad3/BubR1 is a component of the spindle assembly checkpoint, which monitors if the attachment of microtubules to all kinetochores is established or not. When unattached kinetochores exist or tension between sister chromatids is not sufficient, the spindle assembly checkpoint is activated to arrest the cell cycle. Mad2 recognizes unattached kinetochores, and Mad3/BubR1, together with Mad2, binds to Cdc20 to inhibit premature activation of APC/C before anaphase. Although budding yeast Mad3 contains a D-box, Mad3 is not degraded by APC/C-Cdc20. Thus, Mad3 may be a pseudosubstrate inhibitor that docks to the co-activator Cdc20 so that it cannot activate APC/C (Burton and Solomon 2007). In contrast, fission yeast Mad3 seems to be an actual substrate of APC/C to inhibit cyclin degradation, which leads to metaphase arrest if the checkpoint is activated (King et al. 2007).

Mes1

Fission yeast Mes1 is an APC/C inhibitor essential for meiosis I–II transition. Mes1 is a small protein of 11.3 kDa and contains a KEN-box and a D-box to directly bind to the WD40 repeats of an APC/C co-activator Slp1/Cdc20/Fizzy. Mes1 is shown to be an actual substrate of APC/C (Kimata et al. 2008), rather than a pseudosubstrate.

Erp1/Emi2

An Erp1 homolog Emi1 is a pseudosubstrate inhibitor of APC/C. An APC/C inhibitor Erp1 of *Xenopus laevis* has a D-box and a zinc-binding region (ZBR). Mutation in the D-box lost the function of Erp1, and Erp1

may be also a pseudosubstrate inhibitor of APC/C (Nishiyama et al. 2007). Mutation in the ZBR also caused loss of Erp1 function although it still can bind to APC/C, suggesting that dual-inhibition mechanism may operate for Erp1 to inhibit APC/C.

Securin

A recent study found that securin functions not only as a separase inhibitor, but also as an APC/C inhibitor, which assists accumulation of another APC/C substrate Cyclin B to promote mitosis (Marangos and Carroll 2008).

Thus, the molecular mechanisms how APC/C is inhibited can be differentiated depending upon species and/or situations. It is interesting to note that the primary way to block the APC/C activity is to contain the destruction motifs, irrespective of the actual or pseudo substrates.

Cross-References

▶ Meiosis

References

Burton JL, Solomon MJ (2007) Mad3p, a pseudosubstrate inhibitor of APC^{Cdc20} in the spindle assembly checkpoint. Genes Dev 21:655–667

Kimata Y, Trickey M, Izawa D, Gannon J, Yamamoto M, Yamano H (2008) A mutual inhibition between APC/C and its substrate Mes1 required for meiotic progression in fission yeast. Dev Cell 14:446–454

King EM, van der Sar SJ, Hardwick KG (2007) Mad3 KEN boxes mediate both Cdc20 and Mad3 turnover, and are critical for the spindle checkpoint. PLoS ONE 2:e342

Marangos P, Carroll J (2008) Securin regulates entry into M-phase by modulating the stability of cyclin B. Nat Cell Biol 10:445–451

Nishiyama T, Ohsumi K, Kishimoto T (2007) Phosphorylation of Erp1 by p90rsk is required for cytostatic factor arrest in Xenopus laevis eggs. Nature 446:1096–1099

Peters JM (2006) The anaphase promoting complex/cyclosome: a machine designed to destroy. Nat Rev Mol Cell Biol 7:644–656

Rape M, Reddy SK, Kirschner MW (2006) The processivity of multiubiquitination by the APC determines the order of substrate degradation. Cell 124:89–103

Anaplasia

Barbara J. Davis
Section of Pathology, Tufts Cummings School of Veterinary Medicine Biomedical Sciences, North Grafton, MA, USA

Definition

Anaplasdia is the lack of differentiation (to form backward) to represent the extent to which cells within the tumor resemble tissue counterparts.

Cross-References

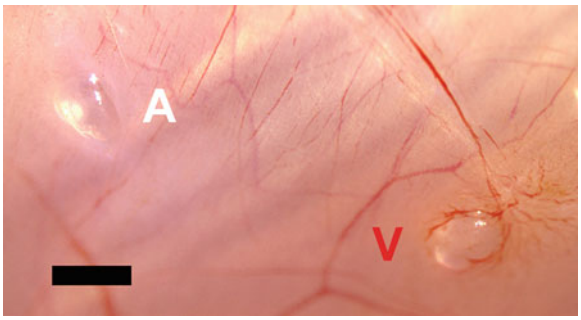
▶ Cancer Pathology

Angiogenic Switch

Marsha A. Moses
Department of Surgery/Harvard Medical School, Vascular Biology Program/Children's Hospital Boston, Boston, MA, USA

Definition

The angiogenic switch refers to a key early stage in solid tumor progression during which an avascular tumor lesion first becomes vascularized (Fig. 1) (Harper and Moses 2006). The acquisition of this angiogenic phenotype is a rate-limiting step in a tumor's development in that these new capillaries provide necessary nutrients and gas exchange for the nascent tumor. The acquisition of a capillary network is required for exponential growth of the tumor and for its metastasis. It has been proposed that the angiogenic switch occurs as a function of a perturbation in the balance between angiogenic stimulators and inhibitors in favor of the positive regulators (Folkman 2002; Harper and Moses 2006).



Angiogenic Switch, Fig. 1 An in vivo tumor model that reliably recapitulates the angiogenic switch (Harper and Moses 2006). Preangiogenic tumor nodules (avascular, A) and angiogenic tumor nodules (vascular, V) are shown. Scale bar = 1 mm

Cross-References

► Regulation of Tumor Angiogenesis

References

Folkman J (2002) Role of angiogenesis in tumor growth and metastasis. Semin Oncol 29(6 Suppl 16):15–18

Harper J, Moses MA (2006) Molecular regulation of tumor angiogenesis: mechanisms and therapeutic implications. In: Cancer: cell structures, carcinogens and genomic instability. Birkhauser, Basel, pp 223–268

Anisocytosis

Barbara J. Davis
Section of Pathology, Tufts Cummings School of Veterinary Medicine Biomedical Sciences, North Grafton, MA, USA

Definition

Anisocytosis – Variation in cell size.

Cross-References

► Cancer Pathology

Anisokaryosis

Barbara J. Davis
Section of Pathology, Tufts Cummings School of Veterinary Medicine Biomedical Sciences, North Grafton, MA, USA

Definition

Anisokaryosis is variation in nuclear size.

Cross-References

► Cancer Pathology

ANOVA

► Analysis of Variance

Anoxia

Bernhard Schmierer
Department of Biochemistry, Oxford Centre for Integrative Systems Biology (OCISB), University of Oxford, Oxford, UK

Definition

Complete absence of oxygen, as opposed to physiologically normal (normoxic) or unphysiologically low (hypoxic) oxygen levels.

Cross-References

► Cell Cycle Signaling, Hypoxia

Antagonist

Melissa L. Kemp
The Wallace H. Coulter Department of Biomedical Engineering, Georgia Institute of Technology and Emory University, Atlanta, GA, USA

Definition

Antagonist is the opposite of an agonist. An antagonist is a ligand that binds to the cell surface receptor but does not trigger any response. Instead, once bound to its target receptor, an antagonist blocks ligand access to the receptor.

Antibody-dependant Immune Response

► B Cell-mediated Immune Response

Antibody-mediated Immunity

► B Cell-mediated Immune Response

Antigen

Gajendra Raghava
Bioinformatics Centre, Institute of Microbial Technology, Chandigarh, Chandigarh, India

Definition

Antigen is a molecule which is recognized as foreign by the host immune system. This may be protein, lipid, or carbohydrate in nature.

Antigen–Antibody Binding Site Prediction

► B Cell Epitope Prediction

Antigen–Antibody Interface Residue Prediction

► B Cell Epitope Prediction

Antigen-Binding Site

► Paratope

Antigenic Determinant

► Epitope
► Systems Immunology, Data Modeling and Scripting in R

Antigenic Drift

► Antigenic Drift and Shift

Antigenic Drift and Shift

Christian Schönbach
Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, Iizuka, Fukuoka, Japan

Synonyms

Antigenic Drift; Antigenic Shift; Change of Antigenic Properties by Mutations; Change of Antigenic Properties by Rearrangement of Viral Genome Segments

Definition

Viral infections result in a constant competition between the virus and host to prevail over each other. The virus tries to avoid or defeat the host defense mechanisms, whereas the host defense tries to eliminate the virus. Two mechanisms enable viruses to

escape the host immune response: antigenic drift and antigenic shift (Weber and Elliott 2002; Boni 2008). RNA viruses (e.g., influenza A) produce their one RNA-dependent RNA-polymerase to replicate their RNA. Since the viral RNA polymerase lacks proofreading functions, sequence mutations occur frequently. If the mutations result in amino acid changes the antigenic properties of the viral protein may change, enabling the virus to escape antibody recognition (antigenic drift) (Boni 2008). Another mechanism that produces viral protein variants with new antigenic properties is the exchange of viral genome segments during mixed viral infections (antigenic shift) (Weber and Elliott 2002).

Cross-References

► Viral Respiratory Tract Infections

References

Boni MF (2008) Vaccination and antigenic drift in influenza. Vaccine 26(Suppl 3):C8–C14

Weber F, Elliott RM (2002) Antigenic drift, antigenic shift and interferon antagonists: how bunyaviruses counteract the immune system. Virus Res 88(1–2):129–136

Antigenic Shift

► Antigenic Drift and Shift

Antimicrobial Peptides

Sneh Lata[1] and Gajendra Raghava[2]
[1]Cold Spring Harbor Laboratory, Genome Research Center, Woodbury, NY, USA
[2]Bioinformatics Centre, Institute of Microbial Technology, Chandigarh, Chandigarh, India

Definition

Antimicrobial peptides (AMPs) are important components of the innate immune system, used by the host to protect itself from different types of pathogenic bacteria. These peptides are gene encoded, evolutionarily conserved, ubiquitous, and produced virtually by all classes of living organisms ranging from bacteria (to exercise competitive exclusion), plants to higher vertebrates (used as weapons to fight a variety of infectious agents). Apart from killing bacteria, these peptides generally show a broad spectrum of activity against fungi, viruses, and even against cancer cells and are considered potential therapeutic agents. Currently, a great deal of interest is shown in antimicrobial peptides, which seem to be promising to overcome the growing problem of antibiotic resistance among bacteria.

Characteristics

Properties

Antimicrobial peptides are very diverse with respect to amino acid sequence and secondary structure (Diamond et al. 2009). Such is the diversity of sequences that even in two closely related species of animals, same peptide sequence is rarely recovered. Despite this diversity, these peptides possess certain conserved features like these are generally 12–100 residues in length, amphipathic in nature, and are predominantly positively charged owing to the abundance of Lysine and Arginine residues. They share certain properties, such as affinity for the negatively charged phospholipids that are present on the outer surfaces of the cytoplasmic membrane of many microbial species. So, on the whole, amphipathicity and net charge are characteristics understandably conserved among many antimicrobial peptides.

Based on their structures, antimicrobial peptides are classified into four major classes: β-sheet, α-helical, loop, and extended peptides (Powers and Hancock 2003). The β-sheet and α-helical classes are most common in nature. Cationic antimicrobial peptides share a common trait: They have the ability to fold into amphipathic or amphiphilic conformations, often induced by interaction with membranes (Powers and Hancock 2003).

Selectivity of Action

A significant consideration in antimicrobial peptide action is the degree to which these peptides distinguishe between microbial and host cells in

settings of potential toxicity. Evidence continues to mount in support that it is the fundamental differences in biochemical and biophysical properties of microbial versus host cells that provide selective toxicity to antimicrobial peptides. The composition of membrane provides an important determinant by which antimicrobial peptides target microbial versus host membranes. Since the surface of the bacterial membranes is more negatively charged than mammalian cells, antimicrobial peptides will show different affinities toward the bacterial membranes and mammalian cell membranes. Cationic property of antimicrobial peptides enables them to interact more effectively with the relatively charged microbial membrane than with the relatively neutral mammalian membranes. So, charge affinity is likely an important means conferring selectivity to antimicrobial peptides. In addition, the presence of cholesterol membrane as stabilizing agent in mammalian cells and its absence in bacterial cells also affect selectivity. Alternatively, access to host tissues may be restricted by the localization and regulated expression of antimicrobial peptides.

Mode of Action

Antimicrobial peptides are believed to have various modes of action to kill bacteria. These cationic peptides first interact with the relatively negatively charged bacterial membranes, attach to them and their amphipathicity allows them to partition thorough the membrane and insert into membrane bilayers to form pores. Pore formation in the cell membrane brings about the leaking of the cytosolic components also known as cytolysis, thus killing the bacteria. Alternatively, these may penetrate the cell membrane and travel inside the cell to bind intracellular molecules like DNA, RNA, and proteins, thus hampering the crucial metabolic activities. However, in many cases, the exact mechanism of killing is not known.

Applications

In the past few decades, antimicrobial peptides have emerged as promising solutions to the problem of growing antibiotic resistance among various strains of bacteria. Researchers are focusing on antimicrobial peptides, which also play an important role in innate immunity, as alternative drugs. Their short length and fast & efficient action against microbes has made them potential candidates as peptide drugs. Unlike conventional antibiotics, antimicrobial peptides act by bacterial membrane disruption, which is absolutely necessary for bacterial physiology; so, the chances of developing resistance are remote. Antimicrobial peptides also possess immunomodulating activity. These peptides have been reported to direct chemotaxis, neutralize Lipopolysaccharides (LPS), and play role in wound healing too. Several peptides and their derivatives have already passed clinical trials successfully (Hancock and Chapple 1999; Levy 2000) and several others are in pipeline as potential therapeutics (Hancock and Chapple 1999). Their role as food preservative is well established; in addition, they have a number of other biotechnological applications, e.g., in veterinary & Medical science, in transgenic plants (Osusky et al. 2000), in aquaculture, and as aerosol spray for patients of cystic fibrosis, as "chemical condoms" to limit the spread of sexually transmitted diseases, can enhance the potency of existing antibiotics in vivo, probably by facilitating access of antibiotics into the bacterial cell (Zasloff 2002). Designing novel peptides with antimicrobial activities requires prediction methods to narrow down the candidate peptides. Computer-aided prediction methods (Lata et al. 2010, 2007) are proving to be of great help in this direction.

Cross-References

- ▶ Antimicrobial Peptides
- ▶ Innate Immunity
- ▶ Pharmaceutical Toxicology, Application of Biosimulation
- ▶ Prediction Model Integration

References

Diamond G, Beckloff N et al (2009) The roles of antimicrobial peptides in innate host defense. Curr Pharm Des 15(21):2377–2392

Hancock RE, Chapple DS (1999) Peptide antibiotics. Antimicrob Agents Chemother 43(6):1317–1323

Lata S, Sharma BK et al (2007) Analysis and prediction of antibacterial peptides. BMC Bioinform 8:263

Lata S, Mishra NK et al (2010) AntiBP2: improved version of antibacterial peptide prediction. BMC Bioinform 11(1):S19

Levy O (2000) Antimicrobial proteins and peptides of blood: templates for novel antimicrobial agents. Blood 96(8):2664–2672

Osusky M, Zhou G et al (2000) Transgenic plants expressing cationic peptide chimeras exhibit broad-spectrum resistance to phytopathogens. Nat Biotechnol 18(11):1162–1166
Powers JP, Hancock RE (2003) The relationship between peptide structure and antibacterial activity. Peptides 24(11):1681–1691
Zasloff M (2002) Antimicrobial peptides of multicellular organisms. Nature 415(6870):389–395

Anti-oncogen

▶ Tumor Suppressor Gene

Antiviral Interferons

▶ Pro-inflammatory Mediators

Apoptosis

Vani Brahmachari and Shruti Jain
Dr. B. R. Ambedkar Center for Biomedical Research, University of Delhi, Delhi, India

Synonyms

Programmed cell death

Definition

The process of death of a cell as an end point of a series of defined ordered biochemical reactions in response to a signal-like DNA damage or activation of certain genes is called apoptosis or programmed cell death. Apoptosis can be contrasted with necrosis which is unordered and unregulated degeneration of cells. Apoptosis does not release toxic by-products unlike necrosis which may produce cellular debris that can damage the surrounding tissue. Whereas, apoptosis involves programmed cell death wherein the cell receives specific signals and all cellular debris are eliminated effectively by exocytosis. During apoptosis cells undergo morphological and biochemical changes that result in cell death. The changes include blebbing; cell shrinkage; nuclear and DNA fragmentation. Apoptosis of tumor cells is beneficial as it terminates the growth of cells on one hand and decreases the chances of metastasis of tumors on the other. On the other hand, excessive and inappropriate cell death is seen in case of neurological diseases such as Parkinson disease.

Cross-References

▶ Epigenetics, Drug Discovery

Application Ontology

▶ Cell Cycle Ontology (CCO)

Applied Natural Language Processing

▶ Applied Text Mining

Applied Text Mining

Kevin Bretonnel Cohen[1] and Karin Verspoor[1,2]
[1]Center for Computational Pharmacology, University of Colorado, Aurora, CO, USA
[2]Victoria Research Laboratory, National ICT Australia, University of Melbourne, Melbourne, VIC, Australia

Synonyms

Natural Language Processing

Definition

Applied text mining involves the application of methods from ▶ natural language processing or other text analysis methodologies to enhance biological data analysis with information extracted from the biological literature.

Characteristics

There are three primary use cases for applied text mining. One of the prototypical use cases in much current work in the field has been the needs of the model organism database curator. These needs may include information retrieval and document classification; gene name detection and mapping to genomic databases; and information extraction, or the automatic extraction of constrained types of facts from text.

Another common use case since the beginning of the modern era in genomic text mining has been database construction. Here the typical model of the task is again information extraction; the goal is to build systems that can assemble facts for a database of very specific types of assertions.

Finally, text mining has been used as a tool for interpreting the results of high-throughput assays. In this use case, users never see the output of text mining itself – rather, text mining becomes an additional knowledge source to use in interpreting a high-throughput data set.

Model Organism Database Curation

Model organism databases, such as Mouse Genome Informatics, curate information from the published literature into databases of the genomes of a specific species. The rate of publication in biomedicine is currently growing exponentially (Hunter and B̃retonnel Cohen 2006), so it is difficult for the model organism databases to manually keep up with the flow of available information. For this reason, model organism databases and associated efforts like the Gene Ontology consortium have been supporters of applied text mining research for some years. This has led to model organism database curators having a large impact on the direction of research in biomedical natural language processing, particularly through their participation in the BioCreative shared tasks (Hirschman et al. 2005; Krallinger et al. 2008b).

Characterizing the workflow of model organism databases in order to best understand the potential insertion points for text mining is a matter of current research. However, it seems clear that a number of practical application types can be of use to database curators.

The most basic of these is a document triage system. The purpose of a document triage system is to take a set of documents and determine which should be read by the database curators and which can safely be ignored. Document triage systems typically assume that there are particular kinds of information whose presence in a document makes it of importance to the model organism database curator. For example, document triage systems have been built to select documents about protein–protein interactions (Krallinger et al. 2008a), the presence of Gene Ontology terms, embryonic development, tumors, and mutant alleles (Hersh and Voorhees 2008). The performance of document triage systems varies widely depending on the application, and it is sometimes difficult to compare between applications because of the use of different metrics to assess their performance.

Once a document has been triaged and selected for the further attention of a model organism database curator, the next most basic practical application type is the gene mention application. Gene mention systems are a type of named entity recognition system (see ▶ Natural Language Processing) whose goal is to recognize gene and protein names in text. Gene mention systems can be used as part of an initial triage step, giving the model organism database curator some indication of what gene or genes are being studied in a paper. They can also be used to assist in reading – when a gene mention system is used to highlight every occurrence of a gene or protein name in a text, it helps to guide the reader's attention to relevant stretches of text. Finally, gene mention can be helpful in selecting sentences from a text that are likely to be of high interest to the model organism database curator. Modern gene mention systems can perform at about 0.90 F-measure (Smith et al. 2008).

It should be noted that it can be useful to find other varieties of named entities in texts, such as Gene Ontology terms. However, work on locating other types of terms is mostly still in its infancy, although considerable success has been reached with some types, such as mutations (Gregory Caporaso et al. 2007). Other semantic types remain a fruitful area for further research.

A closely related problem is gene normalization. Here, the task is to map from some mention of a gene or protein in text to a specific entity in a database. This has been attempted for the Entrez Gene database and for UniProt. It is made difficult by the facts that the same gene name may apply to different organisms and by the fact that the same gene symbol may map to multiple genes in the same organism.

Finally, having selected documents for reading, recognized named entities in text, and normalized those entities to specific database identifiers, the final practical application is to extract information about specific relationships in the text (see ▶ Natural Language Processing). In the simplest example, the model organism database curator might want to be shown interacting proteins. Much more difficult applications exist as well, such as finding relationships between genes and Gene Ontology terms. In any case, any information extraction application for model organism database curation is complicated by the gene normalization problem and normalization of any other entities of interest.

Database Construction

An application since the early days of the modern era in genomic NLP has been database construction. This is approached as an information extraction task. We differ here between model organism database curation, which can be viewed as a type of database construction and the construction of databases that are focused not on a specific organism but on a specific type of phenomenon or class of biological entity. Both rule-based (Blaschke et al. 1999) and machine-learning-based (Craven and Kumlien 1999) approaches have been applied with success. Examples of databases built through text mining include MuTeXt (Horn et al. 2004), a database of mutations in G protein–coupled receptors, and RLIMS-P (Hu et al. 2005), a database of enzymes, their substrates, and the associated phosphorylation sites. Numerous challenges exist for the construction of such systems, including dealing with speculation and negation.

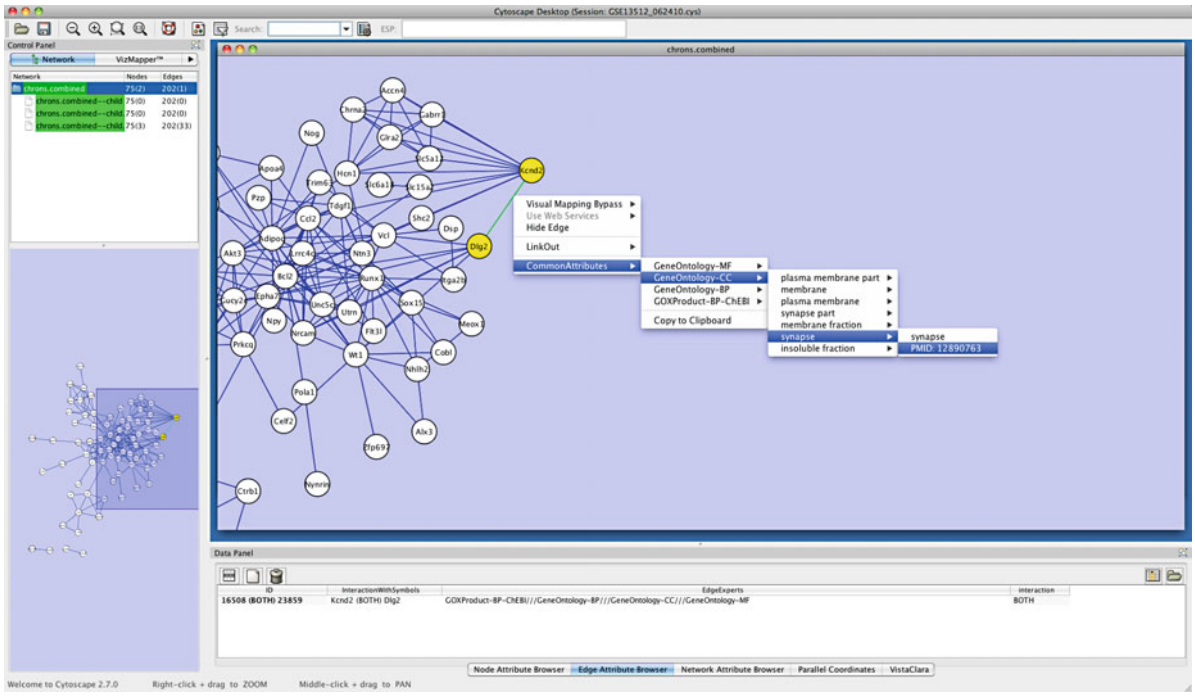
High-Throughput Data Analysis

Unlike the previous use cases, when text mining is used as part of a high-throughput data analysis system, users never see the actual output of the text mining system itself. Early work in this area included using text mining results to improve BLAST searches (Chang et al. 2001) and to analyze gene expression microarray data (Blaschke et al. 2001). These applications are well-reviewed in (Ng 2006). One impressive application is the ChiliBot system (Chen and Sharp 2004). This application is intended for analyzing gene lists from microarray experiments. Given a list of genes, it gathers information from the literature to build a network of associations between those genes. It draws a picture of the network, and for any edge in the network, allows the user to display the set of sentences that provide evidence for that particular edge.

A more recent approach is the Hanalyzer system (Leach et al. 2009). The Hanalyzer has been applied primarily to gene expression data, but is applicable to any data source that can be expressed as a network. It works by building two networks: one showing linked data points in the experimental data (e.g., genes whose expression is significantly correlated), and one showing connections between data points that come from preexisting knowledge sources, such as databases of interacting proteins, known phenotypes, KEGG annotations, and many others. (The Hanalyzer differs from an application like ChiliBot in utilizing these other sources of data.) Overlaying these two networks can both explain observed patterns in the experimental data, and suggest novel hypotheses for further exploration. Where the experimental data overlaps with the data from the knowledge network, the tool provides an explanation for the observed experimental effects and can help the experimentalist both in understanding the experimental data and in the more basic task of validating that the experimental data accords with previous knowledge about the phenomenon under investigation. When the experimental network does not line up with the knowledge network, we see what the new findings of the experiment are, and are suggested with hypotheses for further experimental validation. One such application of the tool led to the discovery of the involvement of four genes in tongue development in the mouse that were not previously known to be involved in this process. The finding was experimentally verified and has furthered our understanding of genetic factors that may lead to the development of cleft palate.

One of the important data sources in the Hanalyzer is text mining output. Figure 1 shows an example of a link in the knowledge network that is obtained through text mining. It shows that the two genes in question are both found in the synapse. As can be seen, the PubMed identifier of the relevant document is given.

The Hanalyzer uses a variety of forms of applied natural language processing. In its simplest form, gene names are analyzed for co-occurrence metrics, using a weighting approach that negates many of the drawbacks of simple co-occurrence. The Hanalyzer also applies a more sophisticated information extraction approach, using the OpenDMAP semantic parser



Applied Text Mining, Fig. 1 Screen capture of the Hanalyzer system, showing the nature of an edge in the graph and the pointer to the PubMed abstract that provides evidence for it

(Hunter et al. 2008). The net effect is a substantial increase in the amount of knowledge available to the system just from preexisting databases.

Software Testing and Quality Assurance in Applied Text Mining

The fact that applied text mining systems are intended to be applied to research in human health and disease places a special burden on system builders to ensure that their systems are built to the highest, industrial-strength standards of software quality. Testing software whose input is language presents special challenges that are not found in other types of software. Here it has been found helpful to combine techniques from software testing with techniques from the field of descriptive linguistics, or the science of describing unknown languages. These two fields turn out to have much in common, both in theoretical background and in terms of practical techniques. In this approach, the text mining application is modeled as a language that we do not know anything about. One finding of this research has been that using structured test suites containing manufactured data can be a more effective tool for uncovering bugs in programs than running those programs against huge test collections, and that in addition, using such test suites is considerably more efficient because it can have a much faster run time (Bretonnel Cohen et al. 2008).

Acknowledgment The authors would like to thank Hannah Tipney for providing the Hanalyzer figure.

Cross-References

▶ Natural Language Processing

References

Blaschke C, Andrade MA, Ouzounis C, Valencia A (1999) Automatic extraction of biological information from scientific text: protein–protein interactions. In: Intelligent Systems for Molecular Biology, AAAI Press, Menlo Park, pp 60–67

Blaschke CJ, Oliveros C, Valencia A (2001) Mining functional information associated with expression arrays. Funct Integr Genomics 1(4):256–268

Bretonnel Cohen K, Baumgartner WA Jr, Hunter L (2008) Software testing and the naturally occurring data assumption in natural language processing. In: Software Engineering, Testing, and Quality Assurance for Natural Language Processing, Association for Computational Linguistics, Columbus, pp 23–30
Chang JT, Raychaudhri S, Altman RB (2001) Including biological literature improves homology search. In: Pacific Symposium on Biocomputing, Mauna Lani, HI, pp 374–383
Chen H, Sharp BM (2004) Content-rich biological network constructed by mining pubmed abstracts. BMC Bioinformatics 5:1471–2105
Craven M, Kumlien J (1999) Constructing biological knowledge bases by extracting information from text sources. In: Intelligent Systems for Molecular Biology, Heidelberg, pp 77–86
Gregory Caporaso J, Baumgartner WA Jr, Randolph DA, Bretonnel Cohen K, Hunter L (2007) Mutationfinder: a high-performance system for extracting point mutation mentions from text. Bioinformatics 23:1862–1865
Hersh W, Voorhees E (2008) TREC genomics special issue overview. Inf Retrieval 12:1
Hirschman L, Yeh A, Blaschke C, Valencia A (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. BMC Bioinformatics 6:S1
Horn F, Lau AL, Cohen FE (2004) Automated extraction of mutation data from the literature: application of MuteXt to G protein-coupled receptors and nuclear hormone receptors. Bioinformatics 20(4):557–568
Hu ZZ, Narayanaswami M, Ravikumar KE, Vijay-Shanker K, Wu CH (2005) Literature mining and database annotation of protein phosphorylation using a rule-based system. Bioinformatics 21(11):2759–2765
Hunter L, Bretonnel Cohen K (2006) Biomedical language processing: what's beyond PubMed? Mol Cell 21:589–594
Hunter L, Lu Z, Firby J, Baumgartner WA Jr, Johnson HL, Ogren PV, Bretonnel Cohen K (2008) OpenDMAP: an open-source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-specific gene expression. BMC Bioinformatics 9(78) http://www.biomedcentral.com/1471-2105/9/78 (doi:10.1186/1471-2105- 9-78)
Krallinger M, Leitner F, Rodriguez-Penagos C, Valencia A (2008a) Overview of the protein–protein interaction annotation extraction task of BioCreative II. Genome Biol 9(suppl 2):S4
Krallinger M, Morgan A, Smith L, Leitner F, Tanabe L, Wilbur J, Hirschman L, Valencia A (2008b) The BioCreative II – critical assessment for information extraction in biology challenge. Genome Biol 9:1
Leach SM, Tipney H, Feng W, Baumgartner WA Jr, Kasliwal P, Schuyler RP, Williams T, Spritz RA, Hunter L (2009) Biomedical discovery acceleration, with applications to craniofacial development. PLoS Comput Biol 5(3):e1000215
Ng S-K (2006) Integrating text mining with data mining. In: Ananiadou S, McNaught J (eds) Text mining for biology and biomedicine. Artech House, Norwood
Smith L, Tanabe LK, Johnson R, Kuo CJ, Chung IF, Hsu CN, Lin YS, Klinger R, Friedrich CM, Ganchev K, Torii M, Liu HF, Haddow B, Struble CA, Povinelli RJ, Vlachos A, Baumgartner WA Jr, Hunter L, Carpenter B, Tsai RTH, Dai HJ, Liu F, Chen YF, Sun CJ, Katrenko S, Adriaans P, Blaschke C, Torres R, Neves M, Nakov P, Divoli A, Mana-Lopez M, Mata J, Wilber WJ (2008) Overview of BioCreative II gene mention recognition. Genome Biol 9(Suppl 2): S2

(Approx.) Boundary Condition

► Constraint

ARC

► Mediator

Archetypes

Jesus Bisbal
Departament de Tecnologies de la Informació i les Comunicacions, Universitat Pompeu Fabra, Barcelona, Spain

Synonyms

Templates

Definition

Archetypes are the formal representation of information concepts used in a given domain of application.

An advanced approach to the modeling and management of information separates domain information models into three independent components: data, archetypes (i.e., structure), and semantics. Archetypes are used to define the different sets of related data attributes, which together provide the necessary context to adequately interpret the information which is to be stored, communicated, or shared. External ontologies or terminological resources are optionally referred to from the archetype definitions in order to annotate those definitions with semantically rich resources, facilitating interoperability and integration.

This approach originated in the medical informatics community to standardize medical information communication, but it is of general applicability. It advocates a principled design methodology to information modeling.

Archetypes are at the core of this methodology, which is often referred to as two-level modeling paradigm (Bisbal and Berry 2011). Its first level, the reference model, is a predefined set of very abstract classes and aggregation rules that provide the flexibility to model any information concept. Its second level, the archetypes, add semantics and constraints to the potential instances of the reference model, in order to ensure that the desired semantics are captured by the clinical concepts defined by those archetypes.

The two-level modeling paradigm is being standardized by the major bodies in medical informatics (Europe's CEN 13606, and USA's HL7 RIM v3). While traditional information modeling methodologies produce large and detailed domain-specific models, the two-level modeling paradigm does not specify a predefined set of attributes that can be represented. It therefore provides a higher level of flexibility which is expected to shield information systems from software maintenance and evolutionary costs.

References

Bisbal J, Berry D (2011) An analysis framework for electronic health record systems: interoperation and collaboration in shared healthcare. Methods Inf Med 50(2):180–189

Area under the ROC Curve

Francisco Melo
Pontificia Universidad Católica de Chile, Santiago, Chile
Millennium Institute on Immunology and Immunotherapy, Santiago, Chile

Synonyms

AUC

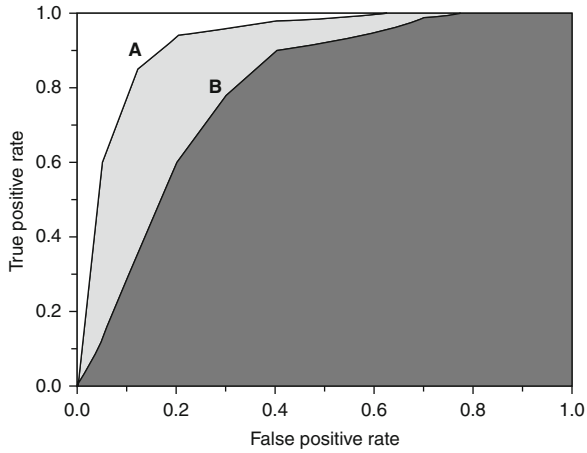
Definition

The area under a ▶ receiver operating characteristic (ROC) curve, abbreviated as AUC, is a single scalar value that measures the overall performance of a binary classifier (Hanley and McNeil 1982). The AUC value is within the range [0.5–1.0], where the minimum value represents the performance of a random classifier and the maximum value would correspond to a perfect classifier (e.g., with a classification error rate equivalent to zero).

The AUC is a robust overall measure to evaluate the performance of score classifiers because its calculation relies on the complete ROC curve and thus involves all possible classification thresholds. The AUC is typically calculated by adding successive trapezoid areas below the ROC curve. Figure 1 shows the ROC curves for two score classifiers A and B. In this example, classifier A has a larger AUC value than classifier B.

The AUC has the important statistical property that it represents the probability that a randomly chosen positive instance will be ranked higher than a randomly chosen negative instance. Therefore, according to this measure, classifier A would have a better classification performance than classifier B.

The ROC curves of two score classifiers are shown. The AUC for classifier B is shown as a dark gray filled area. The AUC for classifier A corresponds to the light gray filled area plus the dark gray filled area.



Area under the ROC Curve, Fig. 1 The AUC is used as an overall measure to compare classifiers

References

Hanley JA, McNeil BJ (1982) The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143:29–36

Argonaute-mediated Cleavage

▶ Target Cleavage

Arrays

▶ DNA Microarrays

Arrow Ontology

▶ Edge Ontology

Artificial Evolution

Christoph Adami
Department of Microbiology and Molecular Genetics, Michigan State University, East Lansing, MI, USA

Synonyms

Simulated evolution

Definition

Artificial evolution refers to any procedure that uses the mechanism of Darwinian evolution to generate a product. While this definition encompasses the evolution of organic life forms by means of artificial selection (breeding of animals, plants, or microorganisms), artificial evolution commonly refers to the instantiation of evolution within a nonbiological medium. Artificial evolution is sometimes used synonymously with ▶ artificial life, but in principle the concepts are overlapping but different.

Characteristics

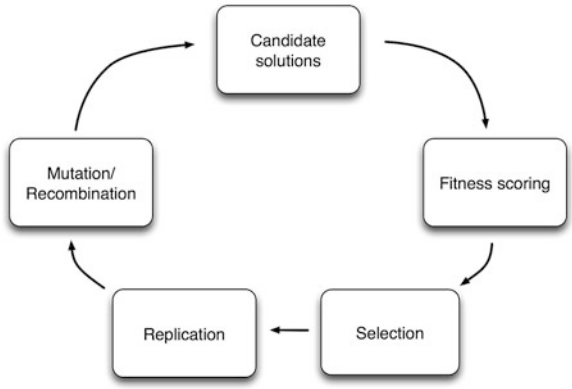
In Computer Science

Within computer science, artificial evolution is commonly implemented in terms of an evolutionary or genetic algorithm (▶ Genetic Algorithms, ▶ Evolution Programming) (GA). A GA is a search procedure that implements the three essential elements of Darwinian evolution - replication, mutation, and selection. Typically, the procedure acts on a population of candidate solutions, which are scored according to a fitness criterion and selected to be replicated according to a formula that takes the fitness score into account (see Fig. 1).

This formula may be as simple as selecting a percentage of the highest fitness individuals (elite selection) or be a probability that is proportional to fitness (roulette wheel selection) or a mixture of those. The fitness function is usually determined by the user so that the maximum of that function is achieved for the optimal solution. In cases where it is not known which function is maximized by the optimal solution to a problem, the construction of an appropriate fitness function can be a difficult research problem. After being selected for replication, individuals are mutated and recombined with a given rate to create new candidate solutions that have inherited the features of their parents but potentially carry new features not previously present in the population. Typically, GAs use a variety of mutation mechanisms to create variation. Evolutionary algorithms are alternatives to more traditional random or heuristic search algorithms and work best in large search spaces where the best solution consists of partial solutions that are also fit.

In Engineering

Artificial evolution can be used to solve engineering problems by applying evolutionary computation techniques to hardware rather than software. For example, it is possible to design electronic circuits in reconfigurable hardware (Thompson 1996) that have novel properties that exploit the material properties of the substrate. In another example of unconventional computing (▶ Unconventional Computation), a team



Artificial Evolution, Fig.1 Typical flow diagram of events in a genetic algorithm

evolved a circuit with the goal of producing an oscillatory signal. One of the resulting circuits evolved a radio receiver that captured the clock signal from a computer on a nearby desk (Bird and Layzell 2002).

In Evolutionary Biology

Artificial evolution is used in evolutionary biology to illustrate the process of evolution itself by instantiating evolution algorithmically, usually within a computer (Adami 1998). One of the earliest uses of computers to study the process of evolution is Richard Dawkins' thought experiment that demonstrates the process of evolution in an artificial medium by evolving the target phrase "Methinks it is like a weasel" from a randomly generated sequence of letters drawn from an alphabet of 28 (Dawkins 1986). In this example, sequences are copied 100 times (instantiating heredity), while 1 in 20 characters is changed (instantiating the process of mutation). The resulting sequences are compared to the target phrase, and the sequence that is closest to the target is chosen to be copied again (instantiating selection), see Fig. 2.

In the same book, Dawkins introduces the "biomorph" program that evolves tree-like structures whose appearance is determined by nine developmental "genes" that determine the tree's appearance. The genes that encode the tree are mutated and recombined at random and selected by the user that, thus, guides the process of evolution to generate desired shapes. Another well-known example of artificial evolution that probes the coevolution of morphology and behavior is due to Karl Sims (Sims 1994), who studied how creatures built from interconnected blocks

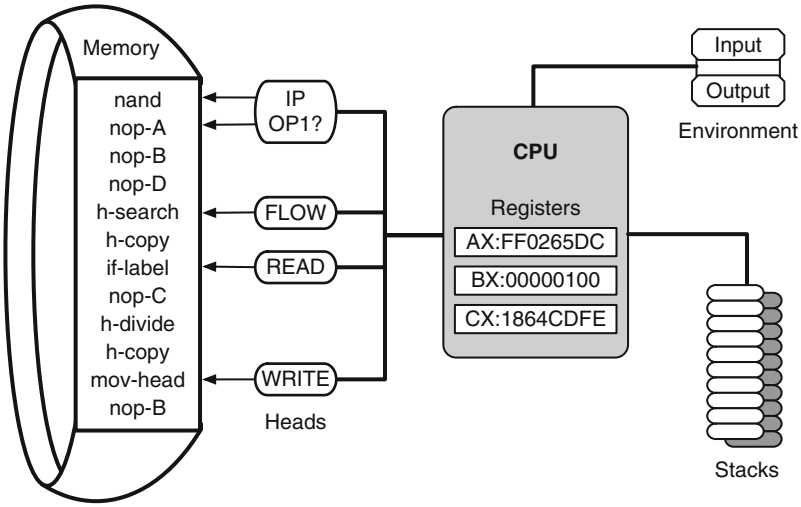
```
Generation 01:  WDLTMNLT DTJBKWIRZREZLMQCO P
Generation 02:  WDLTMNLT DTJBSWIRZREZLMQCO P
Generation 10:  MDLDMNLS ITJISWHRZREZ MECS P
Generation 20:  MELDINLS IT ISWPRKE Z WECSEL
Generation 30:  METHINGS IT ISWLIKE B WECSEL
Generation 40:  METHINKS IT IS LIKE I WEASEL
Generation 43:  METHINKS IT IS LIKE A WEASEL
```

Artificial Evolution, Fig. 2 Selected generations on the line of descent obtained by running Dawkins' "weasel" program

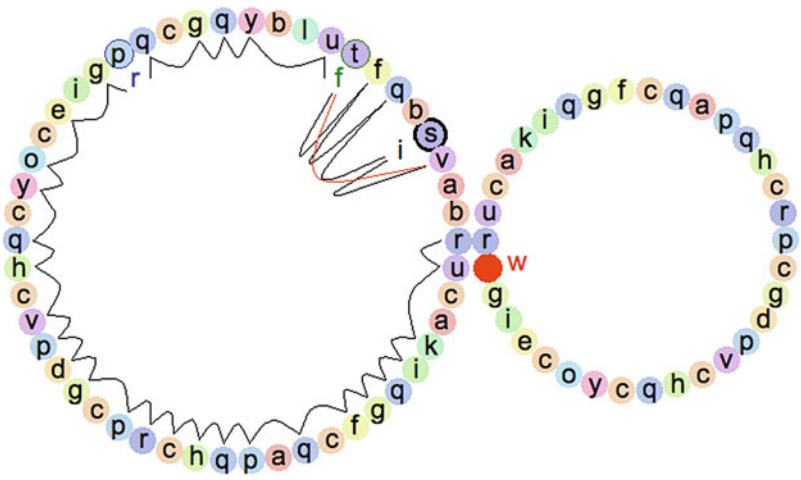
and controlled by a neural architecture evolve in a three-dimensional world with simulated physics. This work highlighted the importance of complex environments in the evolution of complexity.

Digital Life

Computer viruses (or, more generally, malware) evolve by artificial means when the malware creators react to a changed fitness landscape (e.g., new security countermeasures) by adapting their virus to these changes. In this instantiation of artificial evolution, replication is usually a key feature of the malware program, while the fitness function is implicit to the environment within which the virus seeks to replicate, rather than specified externally. Variation is usually directed by the malicious users but can sometimes be autonomous. The concept of an evolving computer program as an instantiation of evolution (as opposed to a simulation of evolution) gave rise to the field of digital life, where self-replicating computer programs are executed on virtual (simulated) central processing units (CPUs). By encasing programs in a virtual world, it is possible to design the language they are written in (their genetic code), to control the program's rate of mutation, and the complexity of the world they live in. The first working digital life system called "tierra" was created by Tom Ray (Ray 1992). In tierra, self-replicating computer programs live in simulated core memory and compete for CPU time and memory space. Because the ▶ fitness of any particular ▶ digital organism in tierra is solely determined by its ability to contribute offspring to future generations, there is no a priori optimal program. The digital life system "Avida" (see Figs. 3 and 4) was developed at the California Institute of Technology in order to study fundamental aspects of evolutionary biology, such as the evolution of complexity (Adami and Brown 1994; Ofria et al. 1998). Digital life research can provide a system's view of evolutionary biology because



Artificial Evolution, Fig. 3 The CPU acting on a self-replicating program in the Avida environment. In this version, four threads (marked "Heads") are executing the avidian code simultaneously, akin to the transcription of genetic code by four DNA polymerases



Artificial Evolution, Fig. 4 An avidian genome in the act of self-replication. Letters *a–z* stand for one of the 26 possible instructions. The *solid black line* follows the forward execution, while *red* denotes a backward jump

digital organisms are best understood within the selective environment they evolved in, which includes the population of programs itself.

Digital life research has given rise to a number of discoveries in evolutionary biology that have been validated later in biological organisms, such as the "survival-of-the-flattest" effect or the importance of ▶ epistasis in the evolution of ▶ complexity (Adami 2006). This type of work illustrates the idea that the difference between evolution in the biochemical realm and evolution in the computational domain lies only in the substrate that carries the information that evolves, but that the processes that underlie the dynamics (the algorithmic rules) are the same. Because of this generality, digital life systems such as Avida have been used not only to study the process of evolution itself but can also be adapted to study the evolution of behavior and intelligence as well as software design (Beckmann et al. 2008).

Cross-References

- ▶ Artificial Life
- ▶ Complexity
- ▶ Digital Organism
- ▶ Epistasis
- ▶ Evolution Programming
- ▶ Fitness
- ▶ Genetic Algorithms
- ▶ Unconventional Computation

References

Adami C (1998) Introduction to artificial life. Springer, New York
Adami C (2006) Digital genetics: unravelling the genetic basis of evolution. Nat Rev Genet 7:109–18
Adami C, Brown CT (1994) Evolutionary learning in the 2D artificial life system "Avida". In: Brooks R, Maess P (eds) 4th international conference on the synthesis and simulation of living systems. MIT Press, Boston, pp 377–381
Beckmann BE, Grabowski LM, McKinley PK, Ofria C (2008) Applying digital evolution to the design of self-adaptive software. In: IEEE symposium on artificial life '09, IEEE Computational Intelligence Society, Nashville, TN, pp 100–107
Bird J, Layzell P (2002) The evolved radio and its implications for modelling the evolution of novel sensors. In: Evolutionary computation, 2002. CEC '02, IEEE Press, pp 1836–1841
Dawkins R (1986) The blind watchmaker. Oxford University Press, Oxford, UK
Ofria C, Brown CT, Adami C (1998) The avida user's manual. In: Adami C (ed) Introduction to artificial life. Springer, New York, pp 297–350
Ray TS (1992) An approach to the synthesis of life. In: Langton CG, Taylor C, Farmer JD, Rasmusse S (eds) Second interdisciplinary workshop on the synthesis and simulation of living systems. Addison Wesley, Santa Fe, p 371
Sims K (1994) Evolving virtual creatures. In: Glassners A (ed) Proceedings of the 21st annual conference on computer graphics and interactive techniques. ACM Press, pp 15–22
Thompson A (1996) Silicon evolution. In: Koza JR, Goldberg DE, Fogel DB, Riolo RL (eds) Genetic programming 1996: proceedings of the 1st annual conference (GP96). MIT Press, pp 444–445

Artificial Immune Systems

► Lymphocyte Dynamics and Repertoires, Modeling

Artificial Life

Jan T. Kim
School of Computing Sciences, University of East Anglia, Norwich, Norfolk, UK

Definition

Artificial Life (Langton 1992b; Bedau et al. 2000) is a highly interdisciplinary field of research which comprises areas of the biological sciences including genetics, evolutionary biology, ecology, neurobiology, and behavioral biology, as well as aspects of computer science, mathematics, engineering, and robotics. Predecessor disciplines that have informed Artificial Life also include theoretical and mathematical biology and cybernetics. In turn, Artificial Life can be one of the predecessors of Systems Biology.

The central aim of Artificial Life is to identify, understand, and use the fundamental principles that underpin and organize biological systems. Special emphasis is given to finding *simple* principles or rules that give rise to the complexity that is characteristic of biological systems, as complexity is regarded as an emergent phenomenon (► Emergence). A central methodological pattern of Artificial Life research is to study dynamics or phenomena of biological systems by synthesizing or simulating them in entirely different media. These media include software (e.g., using computational or mathematical models to study evolution), hardware (e.g., using robots to study behavior), and even wetware (i.e., building systems from scratch using techniques from chemistry and molecular biology).

The Artificial Life approach is analogous to that of Artificial Intelligence, which aims to engineer intelligent machines and comprises various techniques that are based on principles gleaned from existing natural systems (e.g., the brain). The name "Artificial Life" should be understood as a reflection of this analogy.

Interest in the synthesis of life-like systems may derive from two distinct motivations. On the one hand, bio-inspired mechanisms are adopted to engineer systems that exhibit desirable properties of living systems. Examples for such properties include robustness to unpredictable environments, learning, or other forms of adaptation. On the other hand, such systems may be built to be amenable to observation or experimentation that is infeasible with the original object of bioscientific inquiry. Computational simulations are paradigmatic of this approach, as they allow continuous, nondestructive observation of dynamical processes where such observation would be impossible for objects in molecular biology.

The biosciences typically focus on existing living systems, and understanding of biological system therefore tends to be descriptive rather than predictive. The scope of Artificial Life goes beyond the existing "life as we know it," and expressly includes "life as it could be." As an example, a traditional approach to

computational modeling in biology might suggest that parameters, such as a mutation rate, should be based on empirical measurement or observation, whereas it is a typical Artificial Life approach to scan a range of settings for such a parameter. In this regard, the field draws from the formal sciences within its interdisciplinary spectrum and focuses on defining and understanding state spaces, rather than on describing individual states. Artificial Life and Systems Biology share this methodological feature, as well as the underlying vision of developing predictive elements within the biosciences.

Characteristics

Evolution

Evolution has generated the living systems that exist today. Biological evolution is assumed to be open ended: There is no limit to the diversity and complexity of the systems that it generates, and the space of evolutionary search is to be shaped and extended by the evolving systems themselves. ▶ Artificial evolution has been used both as a means to study evolution and to explore possible uses of evolutionary mechanisms for tackling various computing challenges. Effects of evolving systems on the search space are a key prerequisite for evolvability in generalized biology (▶ Evolvability, Generalized Biology).

Computer models of evolution are often structurally similar to ▶ evolutionary algorithms. However, Artificial Life studies tend to place emphasis on modeling biological features of evolution, such as open-endedness, whereas evolutionary algorithms are often developed for optimization purposes. In such applications, the fitness function and its domain are externally defined and static, i.e., the fitness value of an individual does not depend on other individuals. In contrast to this, Artificial Life models of evolution often comprise an intrinsic fitness concept (Bedau and Packard 1992), where fitness of an individual depends on its interactions with others or on an environment that is shaped by the actions of individuals.

In many models, fitness is an implicit or emergent property. For example, in the Tierra (Ray 1992) and Avida (Ofria et al. 1998) systems, programs on a virtual machine are subject to Darwinian evolution. These systems cannot straightforwardly be applied for optimization, but they have been highly successful in capturing phenomena such as the emergence of genetic parasitism.

Substantial research interest has focused on the major transitions in evolution (Smith and Szathmáry 1995), and in particular in the transition from chemical to biological evolution which took place upon the emergence of the first living systems. Hypercycles (Eigen and Schuster 1978), which may have formed in an RNA world (Gesteland et al. 2006), provide an explanation of the emergence of self-replication, biological evolution. They are therefore considered as a key step in the transition to biological evolution.

Hypercycles are based on artificial chemistry-like systems in which molecules catalyze synthesis of other molecules. The product of such synthesis may itself have catalytic activity. A hypercycle is a set molecules in which each molecule catalyzes synthesis of the next one, with the last catalyzing synthesis of the first molecule. Synthesis of molecules in a hypercycle will accelerate more quickly than that of molecules that do not participate in a hypercycle. However, assuming a homogeneous reaction mixture, hypercycles are vulnerable to molecules that are parasitic in the sense that they receive catalytic support but do not provide any such support back to the hypercycle. However, as a key Artificial Life contribution, Boerlijst and Hogewag (1992) have shown that this vulnerability does not necessarily exist in a nonhomogeneous reactor.

While software continues to be a predominant medium for studying evolution, advances in molecular biotechnology have enabled instantiations of evolutionary systems in wetware. Examples include directed and other forms of in vitro evolution (Gesteland et al. 2006), and a study of mutation rate evolution during 10,000 generations of *Escherichia coli* (Sniegowski et al. 1997), illustrating that mutual exchange between experimental wetware approaches and theoretical software-based investigation is a common pattern of Artificial Life and Systems Biology.

Gene Regulation and Gene Regulatory Networks

▶ Gene regulation and the dynamics of gene expression have attracted interest in theoretical biology and cybernetics following the discovery of gene regulatory mechanisms which led to the operon model introduced by Jacob and Monod. Further abstraction from biochemical details and numerical characteristics of individual regulatory mechanisms resulted in various Artificial Life models of gene regulation and

▶ regulatory networks. In the *NK* model of Boolean networks (Kauffman and Weinberger 1989), gene activity is discretized into "off" and "on" states, and each of the *N* genes is regulated by a Boolean function taking its inputs from *K* regulators. Boolean networks have subsequently been generalized to ▶ Probabilistic Boolean Networks (Zhang et al. 2007), which are used in Systems Biology to model gene ▶ regulatory networks.

Numerous approaches to computationally capture gene expression dynamics have been developed in Systems Biology and its forerunning interdisciplinary areas. Integration of regulatory networks into models that also address other biological processes has been a major focus of interest in Artificial Life. As an example, the suggestion (by developmental biologist Lewis Wolpert) that a regulatory network could create positional information and, on this basis, a striped "French flag" pattern has been taken up as a challenge by Artificial Life researchers, and it has resulted in numerous computational models of morphogenetic pattern formation, e.g., by Flann et al. (2005); Knabe et al. (2010). While "French flag" models typically use a static spatial structure, such as a lattice, other models and systems include a dynamic morphogenetic structure which may feed back into gene expression levels (Marnellos and Mjolsness 1998; Kim 2001). Also, evolution of regulatory networks that organize morphogenesis has been studied (Kim 2000). In summary, Artificial Life research has provided insights into various aspects of gene regulatory networks, such as their attractor structure and their evolvability. These systems level properties continue to be relevant to Systems Biology as well.

Development and Morphogenesis

Development of patterns and morphological structures has inspired formal models such as reaction-diffusion systems (Meinhardt 1982) and Lindenmayer systems (Prusinkiewicz and Lindenmayer 1990). Many of these approaches were adopted and further developed within Artificial Life. Kumar and Bentley (2003) provide a selection of computational approaches to morphogenesis.

Models which represent morphologies at a level of physical realism often include further aspects to achieve biological plausibility or realism beyond visual appearance. For example, the model by Sims (1994) comprises virtual creatures with three-dimensional morphological structures, ▶ neural networks, to control their behavior. and an evolutionary component to generate creatures that achieve given tasks. The more recent Framsticks platform (Adamatzky and Komosinski 2005) provides a similar level of physical realism.

▶ Lindenmayer systems (Prusinkiewicz and Lindenmayer 1990) use a small set of simple, local growth rules to simulate plant morphogenesis, thus making them a useful concept to study mechanisms of emergence of morphological complexity. While traditional Lindenmayer systems operate on trees, extensions that operate on general graphs have been developed by Kniemeyer et al. (2004) to enable simulation, e.g., of multicellular structures that are not trees in the graph-theoretic sense. Other extensions of Lindenmayer systems enable their use in studying the interplay of gene expression dynamics and growth (Kim 2001) and simulating evolution of morphology and morphogenesis (Jacob 1996).

Robots and Other Hardwares

Robots and other hardware systems are part of the physical world and therefore subject to physical forces. This can be an essential advantage over software-based virtual systems in which much efforts may be required to simulate physical aspects. Therefore, robots and other physical systems are used in Artificial Life as a medium that is complementary to software-based approaches.

Solving technical problems by drawing inspirations from the biological object of scientific inquiry or inspiration frequently is a prominent motivation underpinning robot based research. As an example, the subsumption architecture for robots (Steels and Brooks 1995) aims to modularize and distribute the control logic within the physical structure of a robot in a bio-inspired way. It also provides a demonstration how biological systems may achieve behavior that is adaptive to their environment without relying on an explicit representation of the environment. Another important concept in this work is embodiment, i.e., the idea that cognitive or behavioral capacities are intimately linked with the physical structure of an agent.

In evolutionary robotics (Nolfi and Floreano 2000), evolutionary algorithms are applied to build controllers of a robot's behavior. In other lines of research, evolutionary concepts are applied in robotics, e.g., to evolve controllers, robot structures, or to jointly evolve both.

In ▶ unconventional computation, evolution is also invoked to design other types of computer hardware.

Swarms and Multi-Agent Approaches

Many biological systems consist of multiple units, e.g., tissues consist of cells. While the individual units, such as single neurons, are relatively simple, complexity results from their collective operation within a system. Systems of this type represent emergence of complexity from simple units and principles, and therefore it is a focus of Artificial Life research.

As a classical example, boids ("bird-oids") (Reynolds 1987; Carlson 2000 (Nov) are software agents follow simple rules of avoiding others, aligning with others, and tending toward the swarm's center of mass for cohesion. These simple behavioral principles result in complex and visually realistic swarm-like behavior. In Systems Biology, multi-agent systems are adopted for modeling intracellular processes and tissues (Kiran et al. 2010).

Self-Organization

Systems that are capable of maintaining their own structure are called self-organizing or autopoietic. Cells, with their ability to synthesize all their constituent components, are paradigmatic of autopoiesis. Within Artificial Life, the concept of ▶ computational autopoiesis has been studied.

Artificial chemistries (Dittrich et al. 2001) are systems comprised of a set of chemical substances and a set of rules that describe reactions in terms of educts and products. They provide a framework for studying the emergence of ▶ organizations. Many computational models of biochemical networks that are used in Systems Biology are structurally closely related to artificial chemistries and, thus, amenable to analysis from a perspective of chemical organization. Hypercycles (see above) are examples of chemical organizations, and their emergence can be studied in artificial chemistry systems.

Computation and Information Theory

Information theory and theory of computation have had a strong influence on Artificial Life. Realizing that self-replication is a fundamental capacity of living systems, John von Neumann designed a logical machine on a lattice automaton which is capable of self-replication and of universal computation. Combining these two key capabilities results in a system of considerable complexity. Langton (1984) showed that by omitting the requirement of universal computation, a much simpler lattice automaton enabling self-replicating structure (sometimes called Langton loops) is feasible. This raised the possibility that once self-replication emerges, e.g., as explained by the theory of the hypercycle, computing capabilities could gradually evolve.

Cellular automata continue to be important to Artificial Life. Langton (1992a) has outlined key conditions for complexity and universal computation in cellular automata. Wuensche (2011) has extensively investigated attractor properties of cellular automata. He has also highlighted the close structural similarity between cellular automata and Boolean network, indicating the potential of discrete dynamics approaches for regulatory network analysis.

Langton (1992a) characterizes living systems as being determined by a dynamics of information, rather than by the dynamics of energy. Consequently, a stream of biological modeling in which information theory (▶ Information Theory and Toponomics) (Cover and Thomas 1991) plays a prominent role has developed within Artificial Life (▶ Information in Biological Modeling). Formal approaches on a behavioral level consider agents which are equipped with sensors and actuators. Sensors provide the agent with information from its environment (e.g., the direction of a target object), and actuators enable the agent to manipulate its environment (e.g., by moving around). Using such a scenario, relevant information provides insights into the amount of information required to complete a given task, and empowerment provides a means of identifying advantageous strategies when there is no known reward function.

Cross-References

- ▶ Artificial Evolution
- ▶ Computational Autopoiesis
- ▶ Emergence
- ▶ Evolutionary Algorithms
- ▶ Evolvability, Generalized Biology
- ▶ Gene Regulation
- ▶ Gene Regulatory Networks
- ▶ Identification of Gene Regulatory Networks, Neural Networks

- ▶ Information in Biological Modeling
- ▶ Information Theory and Toponomics
- ▶ Lindenmayer System
- ▶ Organization
- ▶ Probabilistic Boolean Networks
- ▶ Reaction-Diffusion-Advection Equation
- ▶ Regulatory Networks
- ▶ Unconventional Computation

References

Adamatzky A, Komosinski M (eds) (2005) Framsticks: a platform for modelings, simulating and evolving 3D creatures. Springer, Berlin/Heidelberg, pp 37–66

Bedau MA, Packard NH (1992) Measurement of evolutionary activity, teleology, and life. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds) Artificial life II, vol X of Santa Fe institute studies in the sciences of complexity, proceedings. Addison-Wesley, Redwood City, pp 431–461

Bedau MA, McCaskill JS, Packard NH, Rasmussen S, Adami C, Green DG, Ikegami T, Kaneko K, Ray TS (2000) Open problems in artificial life. Artif Life 6:363–376

Boerlijst M, Hogewag P (1992) Self-structuring and selection: spiral waves as a substrate for prebiotic evolution. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds) Artificial life II, vol X of Santa Fe institute studies in the sciences of complexity, proceedings. Addison-Wesley, Redwood City, pp 431–461

Carlson S (2000) Artificial life. Boids of a feather flock together. Sci Am 283:112–114

Cover TM, Thomas JA (1991) Elements of information theory. Wiley, New York

Dittrich P, Ziegler J, Banzhaf W (2001) Artificial chemistries — a review. Artif Life 7:225–275

Eigen M, Schuster P (1978) The hypercycle. a principle of natural self-organization. Part a: emergence of the hypercycle. Naturwissenschaften 64:541–565

Flann N, Hu J, Bansal M, Patel V, Podgorski G (2005) Biological development of cell patterns: characterizing the space of cell chemistry genetic regulatory networks. In: Capcarrere M, Freitas AA, Bentley PJ, Johnson CG, Timmis J (eds) Advances in artificial life (ECAL 2005), vol 3630, Lecture notes in artificial intelligence. Springer Verlag, Berlin/Heidelberg, pp 57–66

Gesteland RF, Cech TR, Atkins JF (eds) (2006) The RNA World. Cold Spring Harbor Laboratory Press, Cold Spring Harbor. ISBN 0-87969-739-3

Jacob C (1996) Evolution programs evolved. In: Voigt H-M, Ebeling W, Rechenberg I, Schwefel H-P (eds) PPSN-IV. Springer-Verlag, Berlin, pp 42–51

Kauffman SA, Weinberger EW (1989) The NK model of rugged fitness landscapes and its application to maturation of the immune response. J Theor Biol 141:211–245

Kim JT (2000) LindEvol: artificial models for natural plant evolution. Künstliche Intelligenz 1(2000):26–32

Kim JT (2001) Transsys: a generic formalism for modelling regulatory networks in morphogenesis. In: Kelemen J, Sosik P (eds) Advances in artificial life (ECAL 2001), vol 2159, Lecture notes in artificial intelligence. Springer Verlag, Berlin/Heidelberg, pp 242–251

Kiran M, Richmond P, Holcombe M, Chin LS, Worth D, Greenough C (2010) FLAME: simulating large populations of agents on paralled hardware architectures. In: Proceedings of the 9th international conference on autonomous agents and multiagent systems: volume 1, vol 1. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, pp 1633–1636, http://dl.acm.org/citation.cfm?id=1838206. 1838517. ISBN 978-0-9826571-1-9

Knabe JF, Schilstra MJ, Nehaniv C (2010) Evolution and morphogenesis of differentiated multicellular organisms: autonomously generated diffusion gradients for positional information. In: Bullock S, Noble J, Watson RA, Bedau MA (eds) Artificial life XI. MIT Press, Cambridge, MA, pp 321–328

Kniemeyer O, Buck-Sorlin GH, Kurth W (2004) A graph grammar approach to artificial life. Artif Life 10(4):413–431. doi:10.1162/1064546041766451, ISSN 10654–5462

Kumar S, Bentley PJ (eds) (2003) On growth, from and computers. Elsevier, Amsterdam. ISBN 0-12-428765-4

Langton CG (1984) Self-reproduction in cellular automata. Physica D 22:120–149

Langton CG (1992a) Life at the edge of chaos. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds) Artificial life II, volume X of Santa Fe institute studies in the sciences of complexity, proceedings. Addison-Wesley, Redwood City, CA, pp 41–91

Langton CG (1992b) Preface. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds) Artificial life II, volume X of Santa Fe institute studies in the sciences of complexity, proceedings. Addison-Wesley, Redwood City, CA, pp xiii–xviii

Marnellos G, Mjolsness E (1998) A gene network model of resource allocation to growth and reproduction. In: Adami C, Belew RK, Kitanp H, Taylor C (eds) Artificial life VI. MIT Press, Cambridge, MA, pp 433–437

Meinhardt H (1982) Models of biological pattern formation. Academic, London

Nolfi S, Floreano D (2000) Evolutionary robotics. MIT Press, Cambridge, MA. ISBN 978-0-262-14070-6

Ofria C, Brown TC, Adami C (1998) Avida user's manual. In: Adami C (ed) Introduction to artificial life. TELOS/Springer-Verlag, Berlin/Heidelberg

Prusinkiewicz P, Lindenmayer A (1990) The algorithmic beauty of plants. Springer, New York

Ray TS (1992) An approach to the synthesis of life. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds) Artificial life II, volume X of Santa Fe institute studies in the sciences of complexity, proceedings. Addison-Wesley, Redwood City, CA, pp 371–408

Reynolds CW (1987) Flocks, herds, and schools: a distributed behavioral model. Comput Graph 21:25–34, http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/

Sims K (1994) Evolving 3D morphology and behavior by competition. In: Brooks RA, Maes P (eds) Artificial life IV. Cambridge, MA, MIT Press, pp 28–39

Smith JM, Szathmáry E (1995) The major transitions in evolution. Oxford University Press, Oxford, UK. ISBN 978–0198502944
Sniegowski PD, Gerrish PJ, Lenski RE (1997) Evolution of high mutation rates in experimental populations of e. coli. Nature 387:703–705
Steels L, Brooks RA (eds) (1995) The artificial life route to artificial intelligence: building embodied situated agents. Lawrence Erlbaum, New Haven, CT. ISBN 978-0-8058-1518-4
Wuensche A (2011) Esploring discrete dynamics. Luniver Press, Frome, UK
Zhang SQ, Ching WK, Ng MK, Akutsu T (2007) Simulation study in probabilistic boolean network models for genetic regulatory networks. Int J Data Mining Bioinformatics 1:217–240

Asf1

► CIA/Asf1

Assay

► Biological Assay

Association Rule

Johannes Fürnkranz
Technical University of Darmstadt, Darmstadt, Germany

Definition

An association rule is a ► rule where certain properties of the data in the body of the rule are related to other properties in the head of the rule.

A typical application example for association rules are product associations. For example, the rule

$$\text{Bread, butter} \rightarrow \text{milk, cheese}$$

specifies that people that buy bread and butter also tend to buy milk and cheese.

The importance of an association rule is often characterized with two measures:

Support measures the fraction of all rows in the database that satisfy both, body and head of the rule. Rules with higher support are more important.

Confidence measures the fraction of the rows that satisfy the body of the rule, which also satisfy the head of the rule. Rules with high confidence have a higher correlation between the properties described in the head and the properties described in the body.

If the above rule has a support of 10% and a confidence of 80%, this means that 10% of all people buy bread, butter, milk, and cheese together, and that 80% of all people that buy bread and butter also buy milk and cheese.

Cross-References

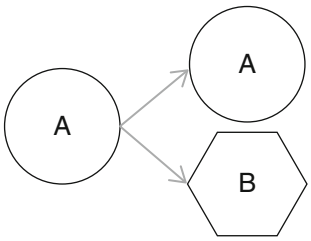
► Rule
► Rule-Based Methods

Asymmetric Cell Division

Heiko Enderling
Center of Cancer Systems Biology, St. Elizabeth's Medical Center - CBR 115D, Tufts University School of Medicine, Boston, MA, USA

Definition

An asymmetric cell division yields two daughter cells with different cellular fate, i.e., a cell of type A gives rise to a daughter cell of type A and a daughter cell of type B.



Cross-References

▶ Cancer Stem Cell Kinetics

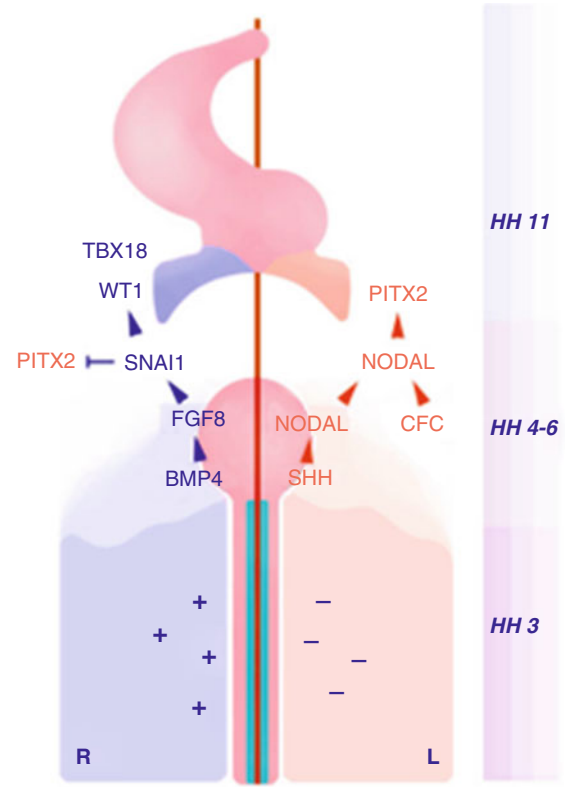
Asymmetry of the Heart, Development

Philippe Huneman
Institut d'Histoire et de Philosophie (IHPST), des Sciences et des Techniques, Université Paris 1 Panthéon-Sorbonne, Paris, France

Definition

A developmental explanation appears as a *cascade* of events involving signaling, production of morphogens, expression, and repression of gene products. Modeling those cascades and identifying the morphogens as well as their propagation dynamics and the cells' receptivity mechanisms yields an explanation of morphogenesis. The embryonic apparition of an asymmetry in the heart results from such cascade, where each moment of the process involves specific genes and molecules, and where important structures are often transitory, a crucial fact first emphasized by advocates of epigenesis such as Caspar Friedrich Wolff or Karl Ernst von Baer.

At the venous pole of the embryonic vertebrate heart, a transitory structure, the proepicardium (PE) produces the epicardium, coronary vasculature, and fibroblasts. In the chicken embryo, the PE displays left-right asymmetry and develops only on the right side, while on the left only a vestigial PE is formed, which subsequently gets lost by apoptosis (Schlueter and Brand 2009). This asymmetry results from a cascade of signaling and gene expression events which involves both sides of the PE, originally symmetrical (Fig. 1). Transitory structures (here PE) proves therefore to be crucial, since the symmetrical form yields the asymmetrical pattern. When the asymmetric heart is built, the PE disappears by being integrated into the structures built (Takano et al. 2007).



Asymmetry of the Heart, Development, Fig. 1 The developmental cascade yielding the LR asymmetry of the cardiac venuous pole

The development of the asymmetry of the heart provides a clear example of our knowledge of specific morphogenesis, its involving various levels of biological hierarchies, and its proper timing.

Cross-References

▶ Explanation, Developmental

References

Schlueter J, Brand T (2009) A right-sided pathway involving FGF8/Snai1 controls asymmetric development of the proepicardium in the chick embryo. PNAS 106(18):7485–7490

Takano K, Ito Y, Obata S, Oinuma T, Komazaki O, Nakamura M, Asashima M (2007) Heart formation and left-right asymmetry in separated right and left embryos of a newt. Int J Dev Biol 51:265–272

Asynchrony

Xiaojuan Sun and Jinzhi Lei
Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University of Beijing, Beijing, China

Definition

Asynchrony, in the general meaning, is the state of not being synchronized (► Synchronization). In signaling network, asynchrony refers to the situation in which each unit in the network shows independent responses to external signals, and no correlation among the units.

Cross-References

► Synchronization

ATAC

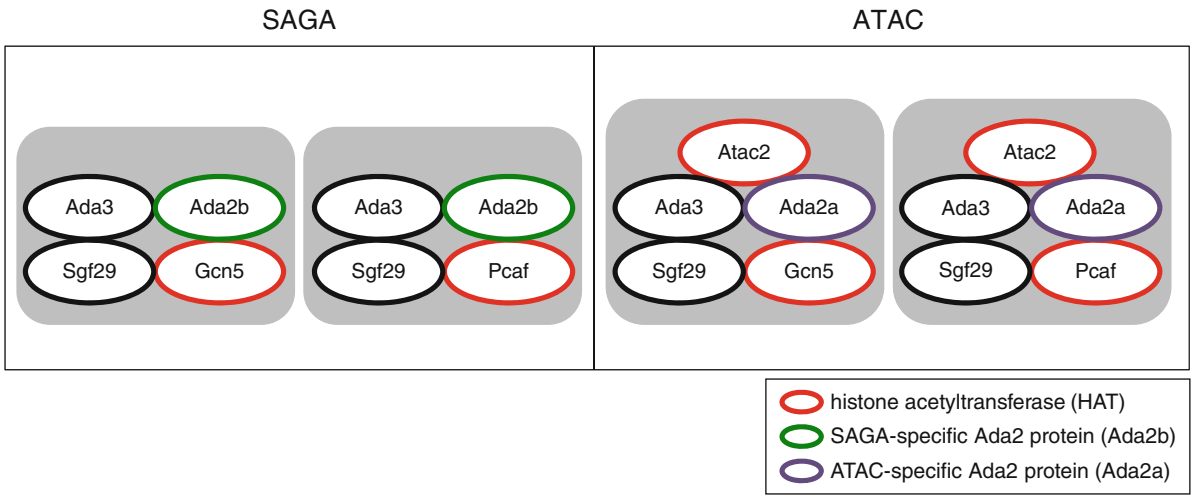
Tetsuro Kokubo
Department of Supramolecular Biology, Graduate School of Nanobioscience, Yokohama City University, Yokohama, Kanagawa, Japan

Synonyms

Ada-Two-A-Containing

Definition

In mammals, yeast SAGA has diverged into two related complexes: SAGA, which like yeast SALSA/SLIK lacks Spt8, and ATAC (*A*da-*T*wo-*A*-*C*ontaining) (Spedale et al. 2012). Human SAGA contains 18 subunits (catalytic subunit Gcn5 or Pcaf, Taf5L, Taf6L, Taf9, Taf10, Taf12, Ada1, Ada2b, Ada3, Spt3, Spt7, Spt20, Sgf11/ATX7L3, Sgf29, Sgf73/ATX7, Usp22, ENY2, and TRRAP), whereas human ATAC contains 12 subunits (catalytic subunit Gcn5 or Pcaf, catalytic subunit Atac2, Ada2a, Ada3, Sgf29, Atac1, Hcf1, Wdr5, NC2β, Yeats2, MBIP, CHRAC17). SAGA and ATAC can both contain either Gcn5 or Pcaf (paralogues) as a catalytic subunit; thus, there are two



ATAC, Fig. 1 Comparison of human SAGA and ATAC

closely related but distinct complexes in each class (Fig. 1). SAGA and ATAC share three subunits (Gcn5 or Pcaf, Ada3, and Sgf29) but contain distinct Ada2 proteins (Ada2b in SAGA and Ada2a in ATAC).

These two HAT complexes are recruited to different genomic loci, where they regulate different sets of genes. Furthermore, SAGA is recruited mainly to the promoter regions of target genes, whereas ATAC is recruited equally to promoter and enhancer regions (Krebs et al. 2011). ATAC binds to enhancers that are not bound by p300, another human HAT. ATAC contains two HAT subunits, Gcn5/Pcaf and Atac2, which confer different substrate specificities: Gcn5/Pcaf acetylates histone H3K9 and H3K14, whereas Atac2 acetylates histone H4K16. The HAT activity of Gcn5 toward mononucleosomes is greatly stimulated when it is in complex with Ada2b-Ada3, but not when it is in complex with Ada2a-Ada3. Thus, the HAT modules in SAGA and ATAC must play different roles in transcriptional regulation. Consistent with this, only SAGA contains the histone H2B deubiquitination (DUB) module (Usp22-Sgf11-Sgf73-ENY2), which appears to regulate the expression of some genes at the elongation stage.

Cross-References

► Mechanisms of Transcriptional Activation and Repression

References

Krebs AR, Karmodiya K, Lindahl-Allen M, Struhl K, Tora L (2011) SAGA and ATAC histone acetyl transferase complexes regulate distinct sets of genes and ATAC defines a class of p300-independent enhancers. Mol Cell 44(3):410–423

Spedale G, Timmers HT, Pijnappel WW (2012) ATAC-king the complexity of SAGA during evolution. Genes Dev 26(6):527–541

ATL

► Adult T-Cell Leukemia

ATLL

► Adult T-Cell Leukemia

ATP-binding Cassette B1 Transporter

Vani Brahmachari and Shruti Jain
Dr. B. R. Ambedkar Center for Biomedical Research, University of Delhi, Delhi, India

Synonyms

ABCB1; MDR1

Definition

It belongs to the family of ATP-binding cassette (ABC) transporters. It is a well-characterized human ABC transporter that was the first of its kind implicated in multidrug resistance of cancer cells. Its normal expression is to prevent the uptake of some lipophilic drugs into the brain and other key organs. It is over-expressed in cancer cells resulting in drugs being pumped out of the cells faster than they can enter. This leads to a lower concentration of the drug in the cell and reduces the effectiveness of the drugs in killing cancer cells.

Cross-References

► Epigenetics, Drug Discovery

ATP-dependent Nucleosome-Remodeling Factors

Toshiya Senda[1] and Naruhiko Adachi[2]
[1]Biomedicinal Information Research Centre (BIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan
[2]Structure-guided Drug Development Project, JBIC Research Institute, Japan Biological Informatics Consortium, Tokyo, Japan

Synonyms

Nucleosome-remodeling factor; Remodeling factor

Definition

ATP-dependent nucleosome-remodeling factors, which belong to the ATP-dependent DNA/RNA helicase superfamily, promote the positional adjustment of nucleosomes with ATP hydrolysis and are involved in cellular processes such as gene expression, genome duplication, repair of damaged DNA, and chromosome recombination (Eberharter and Becker 2004). Several complexes with ATP-dependent nucleosome-remodeling activity have so far been purified; these include SWI/SNF, RSC, NURF, ACF, CHRAC, NuRD/NRD, and INO80 complexes. Biochemical and biological studies have shown that ATP-dependent nucleosome-remodeling factors are large multi-subunit complexes, which are composed of an ATPase subunit and other associated subunits. The ATPase typically belongs to the ATP-dependent DNA/RNA helicase family. ATP-dependent nucleosome-remodeling factors can be categorized by the ATPase subunit into four families, the SWI/SNF, ISW1, CHD, and INO80 families. The ATPase subunit of SWI/SNF and RSC complexes belong to the SWI/SNF family. Since the ATPase subunit of this family contains a bromodomain, discovery of a functional relationship with acetylated histones has been expected. NURF, ACF, and CHRAC complexes possess an ATPase subunit of the ISWI family. The C-terminal region of the ATPase subunit in the ISWI family has a SANT-like domain, which has DNA-binding activity. Indeed, the recent revelation of the crystal structure of ISW1a showed that the SANT domain interacts with DNA and suggested the structure–function relationship of ATP-dependent nucleosome-remodeling factors (Yamada et al. 2011). The NuRD/NRD complex can be categorized into the Mi-2/CHD family. The ATPase subunit of this family contains a chromodomain, which is known to recognize methylated Lys residues in histones. A functional relationship between ATP-dependent nucleosome-remodeling factors and histone methylation has been suggested. The INO80 complex belongs to the INO80 family. The ATPase subunit of the INO80 family contains an insertion of approximately 300 amino acid residues within their ATP-dependent DNA/RNA helicase domain. Functions of the insertion are unknown.

Cross-References

▶ Histone Post-translational Modification to Nucleosome Structural Change

References

Eberharter A, Becker PB (2004) ATP-dependent nucleosome remodeling: factors and functions. J Cell Sci 117:3707–3711

Yamada K, Frouws TD, Angst B, Fitzgerald DJ, DeLuca C, Schimmele K, Sargent DF, Richmond TJ (2011) Structure and mechanism of the chromatin remodelling factor ISW1a. Nature 472:448–453

Attracter

▶ Attractor

Attraction

▶ Attractor

Attractive Force

▶ Attractor

Attractiveness

▶ Attractor

Attractor

Fuyan Hu
Institute of Systems Biology, Shanghai University, Shanghai, China

Synonyms

Attracter; Attraction; Attractive force; Attractiveness; Magnet

Definition

In the study of dynamical systems, an attractor is a "set," "curve," or "space" that is used to describe a system toward which the system tends to evolve regardless of the starting conditions of the system. It is also known as a "limit set." There are five known types of attractors: point attractors, periodic point attractors, periodic attractors, strange attractors, and spatial attractors.

References

Ruelle D. Elements of differentiable dynamics and bifurcation theory. Boston: Academic; 1989. ISBN 0-12-601710-7.

Attribute Selection

► Feature Selection

AUC

► Area under the ROC Curve

Automata

► Modeling Formalisms, Lymphocyte Dynamics and Repertoires

Automated Corpus Generation (CALBC)

Dietrich Rebholz-Schuhmann
European Bioinformatics Institute, Hinxton, UK

Definition

A biomedical corpus contains a large number of biomedical entities that have to be annotated for corpus generation. In the Collaborative Annotation of a Large Biomedical Corpus (CALBC) the named entities from different annotated corpora are harmonized to generate the final corpus.

Introduction

Biomedical text mining (TM) is seeking benchmarks to assess existing TM solutions. A number of challenges have been proposed to achieve this goal: BioCreAtive I and II, JNLPBA and others (Hirschman et al. 2005; Krallinger et al. 2008; Kim et al. 2004, 2009; LLL 05). In all these approaches, the organizers deliver a set of manually annotated documents and ask the challenge participants (CPs) to reproduce the results with their automatic methods.

The first CALBC Challenge is similar in the sense that again an annotated corpus is provided to the challenge participants (CPs) for the reproduction of the annotations, but the first CALBC Challenge is different with regards to the following modifications: (1) the annotated corpus has been generated automatically and not manually (Silver Standard Corpus, SSC-I) and (2) the size of the SSC-I is significantly bigger than the corpora mentioned produced for the other challenges (i.e., 150,000 annotated Medline abstracts for training and testing). The automatic annotation of the corpus has been achieved by automatic harmonization of the contributions from different automatic annotation solutions (Rebholz-Schuhmann et al. 2010). Overall, the proposed approach of the CALBC project can cover a larger number of annotations due to the fact that the annotations are produced automatically and harmonized with automatic means.

Generation of the First and Second CALBC Silver Standard Corpus (SSC-I and SSC-II)

All initial project partners of the CALBC project (PP) annotated the corpus of 150,000 Medline abstracts with their annotation solutions. All annotations were delivered in the IeXML format and concept normalization should make use of standard resources such as UMLS, UniProtKb, and EntrezGene, or should follow the UMLS semantic type system (Rebholz-Schuhmann et al. 2006; Bodenreider and McCray 2003; Bodenreider 2004; The Universal Protein Resource (UniProt) 2009; Maglott et al. 2007). The pair-wise

alignment is based on the methods described in Rebholz-Schuhmann et al. (2010a, b). All contributions from CPs were assessed against the SSC-I by applying exact matching, nested matching, and cosine similarity matching with a 0.98 and 0.9 cosine similarity score (Rebholz-Schuhmann et al. 2010c). All submissions from all participants have been evaluated and the contributions with the best F-measure performance against the SSC-I have been selected for the harmonization into the SSC-II, if the participants did not train their solution on the training data.

Results

All 4 PPs from the CALBC project and in addition, 12 challenge participants (CPs) contributed annotated data sets for evaluation against the SSC-I. CPs could ignore the training data and deliver the annotations from their annotation system, or could train a machine-learning approach on the provided pre-annotated data. In general, the performances of the annotation solutions were lower for the CHED and PRGE in comparison to the identification of DISO and SPE. One explanation is that the terminological resources for DISO and SPE are better standardized as well as the mention of DISO and SPE in the literature.

The best performance over all semantic groups were achieved from two annotation solutions that have been trained on the SSC-I. The performances of the participants' solutions were again measured against the SSC-II and with findings similar to the measurements against the SSC-I. For PRGE, a drop in recall was noticed for the PPs solutions in combination with an increase in recall for the CPs solutions.

Discussion

The manual analysis of the SSC-I and the SSC-II is ongoing work. Due to the size of the corpus, it requires special IT solutions to oversee the regularities and irregularities in the corpus. A selection of irregularities result from the methods applied. First, a number of annotations are not captured ("false negatives," FN, reduced recall) if none of the solutions identifies the entities. An increasing number of contributing annotation solutions reduces the risk that annotations are missed: A bigger number of included annotation solutions lead to a bigger number of annotations that are captured. This achievement is counterbalanced by the number of agreements that have to be available at minimum to accept an annotation.

Second, for the same type of entity, e.g., "insulin," different annotation solutions use a different tag, e.g., PRGE instead of CHED and vice versa. The harmonization of the corpus can account for this, but will not produce this type of polysemous annotation throughout the whole corpus, since not all mentions have been consistently annotated with the two different groups over the whole corpus.

Third, inflections of terms, e.g., tumor vs. tumors and "bear" versus "bearing," lead to disagreements between the different annotation solutions. In the first case, the inflectional variability could be resolved and would lead to higher agreement, in the second case assumptions about the usage of the verb or noun have to be made to resolve conflicts.

Conclusions

The SSC-I delivers a large set of annotations (1,121,705) for a large number of documents (100,000 Medline abstracts). The annotations cover four different semantic groups and are sufficiently homogeneous to be reproduced with a trained classifier leading to an average F-measure of 85%. Benchmarking the annotation solutions against the SSC-II leads to better performance for the CPs' annotation solutions in comparison to the SSC-I.

Cross-References

- BioCreative II.5 and the FEBS Letters Experiment on Structured Digital Abstracts
- BioNLP Shared Task
- Named Entity Recognition
- Natural Language Processing
- Text Corpora, Relationship Extraction

References

Bodenreider O (2004) The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res 32(Database issue):D267–D270

Bodenreider O, McCray A (2003) Exploring semantic groups through visual approaches. J Biomed Inform 36(6):414–432
Hirschman L, Yeh A, Blaschke C, Valencia A (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. BMC Bioinform 6(1):S1
Kim J.D, Ohta T, Tsuruoka Y, Tateishi Y, Collier N (2004) Introduction to the bio-entity recognition task at JNLPBA. In: Proceedings of the JNLPBA-04, Geneva, Switzerland, pp 70–75
Kim J.D, Ohta T, Pyysalo S, Kano Y, Tsujii J (2009) Overview of BioNLP'09 shared task on event extraction. In: Proceedings of the workshop on BioNLP: shared task, Colorado, pp 1–9
Krallinger M, Morgan A, Smith L, Leitner F, Ta-nabe L, Wilbur J, Hirschman L, Valencia A (2008) Evaluation of textmining systems for biology: overview of the second BioCreAtIvE community challenge. Genome Biol 9(2):S1
LLL'05 challenge. http://www.cs.york.ac.uk/aig/lll/lll05/
Maglott D, Ostell J, Pruitt KD, Tatusova T (2007) Entrez gene: gene-centered information at NCBI. Nucleic Acids Res 35 (Database issue):D26–D31
Proceedings of the First CALBC Workshop. http://www.ebi.ac.uk/Rebholz-srv/CALBC/docs/1stworkshopProceeding.pdf
Rebholz-Schuhmann D, Kirsch H, Nenadic G (2006) IeXML: towards a framework for interoperability of text processing modules to improve annotation of semantic types in biomedical text. In: Proceedings of BioLINK, ISMB 2006, Fortaleza, Brazil
Rebholz-Schuhmann D, Jimeno Yepes AJ, van Mulligen E.M, Kang N, Kors J, Milward D, Corbett P, Buyko E, Tomanek K, Beisswanger E, Hahn U (2010) The CALBC silver standard corpus for biomedical named entities: a study in harmonizing the contributions from four independent named entity taggers. In: Proceedings of the LREC (to appear)
Rebholz-Schuhmann D et al (2010b) Assessment of NER solutions against the first and second CALBC Silver Standard Corpus. Semantic Mining in Biomedicine, Hinxton
Rebholz-Schuhmann D, Jimeno Yepes AJ, Van Mulligen EM, Kang N, Kors J, Milward D, Corbett P, Buyko E, Beisswanger E, Hahn U (2010c) CALBC silver standard corpus. J Bioinform Comput Biol 8(1):163–179
The Universal Protein Resource (UniProt) (2009) Nucleic Acids Res 37(Database):D169–D174

Automated Reasoning

C. Maria Keet
KRDB Research Centre, Free University of Bozen-Bolzano, Bolzano, Italy

Definition

Automated reasoning is the approach of implementing the ability to make inferences on a computing system. A formal language is designed in which a problem's assumptions and conclusion can be written down and algorithms are provided to solve the problem with a computer in an efficient way.

Characteristics

People employ reasoning informally by taking a set of premises and somehow arriving at a conclusion, that is, it is *entailed by* the premises (▶ Deduction), arriving at a hypothesis (▶ Abduction), or generalizing from facts to an assumption (▶ Induction). Mathematicians and computer scientists developed ways to capture this formally with logic languages to represent the knowledge and rules that may be applied to the axioms so that one can construct a *formal proof* that the conclusion can be derived from the premises. This can be done by hand (Solow 2005) for small theories, but this does not scale up when one has, say, 80 or more axioms even though there are much larger theories that require a formal analysis, such as checking that the theory can indeed have a model and thus does not contradict itself. To this end, much work has gone into automating reasoning.

The remainder of this section introduces briefly several of the many purposes and usages of automated reasoning, the principal mechanisms (being deduction and to a lesser extent abduction and induction), its limitations, and types of automated reasoners.

Purposes

Automated reasoning, and deduction in particular, has found applications in "everyday life." A notable example is hardware and (critical) software verification, which gained prominence after Intel had shipped its Pentium processors with a floating point unit error in 1994 that lost the company about $500 million. Since then, chips are routinely automatically proven to function correctly according to specification before taken into production. A different scenario is scheduling problems at schools to find an optimal combination of course, lecturer, and timing for the class or degree program, which used to take a summer to do manually but can now be computed in a fraction of it using constraint programming. Much closer to systems biology is the demonstration of discovering (more precisely: deriving) novel knowledge about protein phosphatases by Wolstencroft and coauthors

(in Baker and Cheung 2008). They represented the knowledge about the subject domain of protein phosphatases in humans in a formal ▶ bio-ontology and classified the enzymes of both human and the fungus *Aspergillus fumigatus* using an automated reasoner, which showed that (1) the reasoner was as good as human expert classification, (2) it identified additional p-domains (an aspect of the phosphatases) so that the human-originated classification could be refined, and (3) it identified a novel type of calcineurin phosphatase like in other pathogenic fungi. The fact that one can use an automated reasoner (in this case: deduction, using a Description Logics knowledge base) as a viable method in science is an encouragement to explore such avenues further.

Basic Idea

Essential to automated reasoning are:

1. The choice of the class of problems the software program has to solve, such as checking the consistency of a theory (i.e., there are no contradictions) or computing a classification hierarchy of concepts subsuming one another based on the properties represented in the logical theory
2. The formal language in which to represent the problems, which may have more or less features to represent the subject domain knowledge, such as cardinality constraints (e.g., a member of the arachnids has as part exactly eight legs), probabilities, or temporal knowledge (e.g., that a butterfly is a transformation of a caterpillar)
3. The way how the program has to compute the solution, such as using natural deduction or resolution
4. How to do this efficiently, be this achieved through constraining the language into one of low complexity or optimizing the algorithms to compute the solution or both

Concerning the first item, with a *problem* being, for example, "find out if my theory is consistent," then the *problem's assumptions* are the axioms in the logical theory and the *problem's conclusion* that is computed by the automated reasoner is a "yes" or a "no" (provided the language in which the assumptions are represented is decidable and thus guaranteed to terminate with an answer). With respect to how this is done (item 3), two properties are important for the calculus used: soundness and completeness ($T \vdash \varphi$ if and only if $M \models \varphi$). If it is incomplete, then there exist entailments that cannot be computed (hence, "missing" some results); if it is unsound, then false conclusions can be derived from true premises, which his even more undesirable (Hedman 2004; Portoraro 2010).

An example is included in ▶ Proof, proving the validity of a formula (the class of the problem) in propositional logic (the formal language) using tableau reasoning (the way how to compute the solution) with the Tree Proof Generator (the automated reasoner).

Deduction

Deduction is a way to ascertain if a theory T represented in a logic language entails an axiom α that is not explicitly asserted in T (written as $T \models \alpha$), that is, α can be *derived* from the premises though repeated application of *deduction rules* for instance, a theory that states that "each arachnid has as part 8 legs" and "each tarantula is an arachnid," then one can deduce – it is entailed in the theory – that "each tarantula has as part 8 legs." An example is included in ▶ deduction, which formally demonstrates that a formula is entailed in a theory T using said rules.

Thus, strictly speaking, a deduction does not reveal *novel* knowledge but only that what was already represented implicitly in the theory. Nevertheless, with large theories, it is often difficult to oversee all implications of the represented knowledge, and, hence, the deductions may be perceived as novel from a domain expert perspective, such as with the example about the protein phosphatases. This is in contrast to ▶ abduction and ▶ induction, where the reasoner "guesses" knowledge that is not already entailed in the theory.

Abduction

Compared to deduction, there is less permeation of automated reasoning for abduction. From a scientist's perspective, automation of abduction may seem appealing because it would help one generate a hypothesis based on the facts put into the reasoner (Aliseda 2004). Practically, it has been used for, for instance, fault detection: given the knowledge about a system and the observed defective state, find the likely fault in the system.

To formally capture theory with assumptions and facts and find the conclusion, several approaches have been proposed, each with their specific application areas, for instance, sequent calculus, belief revision, probabilistic abductive reasoning, and ▶ Bayesian networks.

Induction

Logic Induction allows one to arrive at a conclusion that actually may be false even though the premises are true. The premises provide a *degree of support* so as to infer *a* as an explanation of *b*. Such a "degree" can be based on probabilities (a statistical syllogism) or analogy. For instance, we have a premise that "the proportion of bacteria that acquire genes through horizontal gene transfer is 95%" and the fact that "*Staphylococcus aureus* is a bacterium," then we induce that the probability that *S. aureus* acquires genes through horizontal gene transfer is 95%. Induction by analogy is weaker version of reasoning, in particular in logic-based systems, and yields very different answers than deduction. For instance, let us encode that some instance, say, Tibbles is a cat and we know that all cats have the properties of having a tail, four legs, and are furry. When we encode that another animal, Tib, who happens to have four legs and is also furry, then by inductive reasoning by analogy, we conclude that Tib is also a cat, even though in reality it may well be an instance of cheetah. On the other hand, by deductive reasoning, Tib will not be classified as being an instance of cat but may be an instance of a superclass of cats (e.g., still within the suborder *Feliformia*), provided that the superclass has declared that all instances have four legs and are furry. Given that humans do perform such reasoning, there are attempts to mimic this process in software applications, most notably in the area of machine learning and inductive logic programming (Lavrac and Dzeroski 1994). The principal approach with inductive logic programming is to take as input positive examples + negative examples + background knowledge and then derive a hypothesized logic program that entails all the positive and none of the negative examples.

Limitations

While many advances have been made in specific application areas, the main limitation of the implementations is due to the computational complexity of the chosen representation language and the desired automated reasoning services. This is being addressed by implementations of optimizations of the algorithms or by limiting the expressiveness of the language, or both. One family of logics that focus principally on "computationally well-behaved" languages is Description Logics, which are decidable fragments of first order logic (Baader et al. 2008); that is, they are languages such that the corresponding reasoning services are guaranteed to terminate with an answer. Description Logics form the basis of most of the Web Ontology Languages OWL and OWL 2 and are gaining increasing importance in the semantic web applications area. Giving up expressiveness, however, does lead to criticism from the modelers' community, as a computationally nice language may not have the features deemed necessary to represent the subject domain adequately.

Tools

There are many tools for automated reasoning, which differ in which language they accept, the reasoning services they provide, and, with that, the purpose they aim to serve.

There are, among others, generic first- and higher-order logic theorem provers (e.g., Prover9, MACE4, Vampire, HOL4); SAT solvers that compute if there is a model for the formal theory (e.g., GRASP, Satz); constraint satisfaction programming for solving, for example, scheduling problems and reasoning with soft constraints (e.g., Eclipse); DL reasoners that are used for reasoning over OWL ontologies using deductive reasoning to compute satisfiability, consistency, and perform taxonomic and instance classification (e.g., Fact++, RacerPro, Hermit, CEL, QuOnto); and inductive logic programming tools (e.g., PROGOL and Aleph).

Cross-References

- ▶ Abduction
- ▶ Bio-Ontologies
- ▶ Classification
- ▶ Closed World Assumption
- ▶ Deduction
- ▶ Fuzzy Logic
- ▶ Induction
- ▶ Knowledge Representation
- ▶ Open World Assumption
- ▶ Proof, Logic

References

Aliseda A (2004) Logics in scientific discovery. Found Sci 9:339–363

Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (eds) (2008) The description logics

handbook: theory and applications, 2nd edn. Cambridge University Press, Cambridge, UK
Baker CJO, Cheung H (eds) (2008) Semantic web: revolutionizing knowledge discovery in the life sciences. Springer, New York
Hedman S (2004) A first course in logic – an introduction to model theory, proof theory, computability, and complexity. Oxford University Press, Oxford, UK
Lavrac N, Dzeroski S (1994) Inductive logic programming: techniques and applications. Ellis Horwood, New York
Portoraro F (2010) Automated reasoning. In: Zalta E (ed) Stanford encyclopedia of philosophy. Stable URL, http://plato.stanford.edu/entries/reasoning-automated/
Solow D (2005) How to read and do proofs: an introduction to mathematical thought processes, 4th edn. Wiley, Hoboken

Automated Term Recognition

Sophia Ananiadou
National Centre for Text Mining, School of Computer Science, University of Manchester, Manchester, UK

Synonyms

Term extraction; Terminology management

Definition

Automatic term recognition (ATR) refers to the automatic extraction of technical terms from domain-specific texts. It additionally encompasses the assignment of semantic categories of the extracted terms and mapping of them to concepts contained within terminological or ontological resources.

Characteristics

The Need for Term Recognition

With an overwhelming number of textual resources becoming available in biomedicine, there is an increased interest in ▶ text mining techniques that can identify, extract, manage, and exploit biomedical knowledge hidden in the literature (Ananiadou and McNaught 2006). In systems biology, terms are the backbone of such knowledge, since they constitute the linguistic realization of specialized concepts in the biology domain. As the main purpose of terms is to classify scientific knowledge, they are used as a means of scientific communication to convey biological concepts. Once extracted, they can be used to construct conceptual networks of knowledge, and may be organized into hierarchies denoting different types of relationships, such as *is-a*, *part-of*, etc., in addition to domain-specific relations such as *inhibits*, *induces*, etc. These relationships can be subsequently used to develop ontologies (▶ Ontologies). Biological terms are complex, since they include a vast number of synonyms and variants, such as acronyms. As an example, consider the term *ERK2*, which denotes a protein (▶ Named Entity) in *Homo sapiens*. Within the literature, a large number of synonymous term variants of *ERK2* can occur, including the following: *MAPK1, MAP kinase 1, ERT1, extracellular signal-regulated kinase 2, ERK-2, Mitogen-activated protein kinase 2, MAP kinase 2, MAPK2* (http://www.uniprot.org/uniprot/P28482). Addressing terminological variability is crucial to ensure the accuracy of text-based search systems in biology, to allow integration of different resources, and to facilitate the development of terminologies and ontologies (▶ Ontologies). Existing terminological and ontological resources do not cover all instances of terms found in the literature. Since most of these resources are manually curated, keeping track of all new terms and variants that appear in the literature is a virtually impossible task. Term extraction tools can be a great asset to the curators of biological databases and terminological resources, in that they can significantly reduce the burden of keeping the resources up-to-date. The challenges faced by such tools thus include not only the automatic identification of biological terms, but also the identification of synonyms belonging to the same concept, and the disambiguation of terms that belong to different concepts. In order to address all of these challenges, integrated techniques such as automatic term recognition, ▶ term normalization and ▶ term disambiguation are essential for the systematic collection and update of terminologies and their integration into systems biology applications.

Methods of Automatic Term Recognition (ATR)

The process of ATR consists of three steps: term recognition, term classification, and term mapping.

Term recognition identifies terms from the literature. *Term classification* assigns coarse-grained semantic tags to the recognized terms, for example, metabolites, chemical compounds, genes, proteins, etc. These tags are determined by using ▶ named entity recognition techniques. Lastly, the recognized terms are *mapped* to terminological or ontological resources such as UMLS (Bodenreider 2004), OBO (Open Biological Ontologies (http://www.obofoundry.org/)), Uberon (http://obofoundry.org/wiki/index.php/UBERON:Main_Page), GO (Ashburner et al. 2000), and nomenclatures such as HUGO.

The main approaches to term recognition are *dictionary-based*, *statistical*, and *hybrid*, the latter of which combines linguistic knowledge with statistics.

Dictionary-based approaches use existing terminological resources or ontologies to identify terms in text. If a particular sequence of words in a text matches an entry in one of the existing resources, then it is considered to be a term. Although there are several inventories of terms in biology, for example, ChEBI (de Matos et al. 2010), BioThesaurus (Liu et al. 2006), NCBI taxonomy, etc., the use of such resources to aid term recognition has several drawbacks, for example, each resource has a different focus, they are often not linked together, and they do not have sufficient terminological coverage. As new terms are created daily in biology, and as most resources are manually curated and are costly to build, term extraction tools are crucial for populating these resources. Wide coverage terminological resources, such as the BioLexicon (Thompson et al. 2011), can be helpful in the construction of these tools. The BioLexicon combines together terms from a number of existing resources, which are further augmented with terms extracted through the automatic analysis of MEDLINE abstracts, in order to account for real, observed usage of terms in text. The resource incorporates gene/protein term variants and other linguistic and semantic information required to drive text mining applications, such as ▶ information extraction and ▶ information retrieval.

Statistical approaches are based on various statistical distributions of words in text. Since the vast majority of terms consist of nouns or sequences of nouns, the main strategy is to extract specific noun phrases as term candidates and estimate their probability of being terms. The frequency with which sequences of words co-occur in a document or collection of documents that deal with a specific domain can denote their degree of termhood (Kageura and Umino 1996), that is, the extent to which such sequences represent domain-specific terms. Another way of looking at word distribution is to measure the attachment strength between the components of a term (unithood). Techniques like ▶ mutual information, log-likelihood ratios test, frequency of occurrence and likelihood have all been used to measure word distribution. These approaches are typically knowledge poor, that is, they do not make use of existing, domain-specific resources, and are thus portable to many domains. A popular approach used in biology, based on the notion of termhood, is C-value (Frantzi et al. 2000). This is a *hybrid* measure, as it combines linguistic knowledge with a number of statistical characteristics. This approach facilitates the recognition of embedded subterms, which are very frequent in biology. C-Value is the measure underlying the term extraction service TerMine (http://www.nactem.ac.uk/software/termine/), provided by the National Centre for Text Mining.

Incorporating Term Variability and Disambiguation into Term Extraction

The integration of techniques to recognize biological terms and their variants into text search engines improves accuracy, since all possible variants of a particular term can be mapped to the same canonical entity. This means that if user provides any single variant of a term as input to the engine, documents containing all known variants of the term will be returned. Orthographic variations are one of the common types of term variation (e.g., *tumour* vs *tumor*; *NF-KB* vs *NF-kb*; *amino-acid* vs *amino acid*). However, one of the most prolific types of terminological variation by far in biology is acronyms. By way of an example, the acronym *ER* has about 80 possible definitions (expansions), including *estrogen receptor, emergency room, receptor alpha, enhancement ratio*, etc. In turn, each expansion can have several different variant forms. For example, variants of *estrogen receptor* include the following: *oestrogen receptor, estrogen receptors, estrogen-receptor, estrogenic receptor*, etc. Techniques for the automatic extraction of acronyms and their expanded forms are useful for ▶ information retrieval and ▶ information extraction. For example, Wren and Garner (2002) reported that PubMed could retrieve 5,477 documents using the

acroymn *JNK* as a search term, but only 3,773 documents when using the expanded form as a search term, that is, *c-jun N-terminal kinase*. As with term extraction, a common approach for extracting acronyms is utilizing statistical clues in text, that is, co-occurrences between short-forms (e.g., *ER*) and long-forms (e.g., *estrogen receptor*). The strength of these co-occurrences can be measured via techniques such as ▶ mutual information, Dice coefficient, log-likelihood ratio, etc. An example of a service that automatically recognizes acronyms, and expands them into their definitions and variant forms is Acromine (http://www.nactem.ac.uk/software/acromine/).

Term variation is not the only challenge for automatic term management. Terms are frequently associated with multiple meanings. For example, the abbreviation *PCR* has 129 expanded forms that can be consolidated to 30 senses (e.g., *polymerase chain reaction*, *pathologic complete response*, and *phosphocreatine*). In general, each sense has more than one surface form (i.e., variant). Abbreviation disambiguation methods use resources such as Sense Inventories (Okazaki et al. 2010) and apply ▶ clustering to the expanded definitions, using a number of similarity methods to capture the context in which expanded forms appear.

Conclusion

Given the frequency with which new terms appear in the literature, it is necessary to use techniques such as the recognition of terms and their variants, together with disambiguation. These techniques can facilitate the automatic extraction and classification of new terms, and allow them to be linked with databases, ontologies, and controlled vocabularies. An immediate application of automatic term extraction is to enhance the performance of text search engines. While existing search methods normally restrict retrieval to exact matching of query terms, which can result in either low precision (irrelevant information is retrieved) or low recall (relevant information is overlooked), term recognition and term management methods can improve search results by linking terms as they appear in text and in various biological resources, discovering relationships between terms and improving classification and clustering methods.

Cross-References

- ▶ Applied Text Mining
- ▶ Clustering
- ▶ Information Extraction
- ▶ Information Retrieval
- ▶ Mutual Information
- ▶ Named Entity
- ▶ Named Entity Recognition
- ▶ Term Disambiguation, Text Mining
- ▶ Term Normalization, Text Mining

References

Ananiadou S, McNaught J (eds) (2006) Text mining for biology and biomedicine. Artech House, Boston/London

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) Gene ontology: tool for the unification of biology. Nat Genet 25(1):25–29

Bodenreider O (2004) The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Res 32(Database issue):D267–D270

de Matos P, Alcantara R, Dekker A, Ennis M, Hastings J, Haug K, Spiteri I, Turner S, Steinbeck C (2010) Chemical entities of biological interest: an update. Nucleic Acids Res 38(Database issue):D249–D254

Frantzi K, Ananiadou S, Mima H (2000) Automatic recognition of multi-word terms: the C-value/NC-value method. Int J Digit Libr 3(2):115–130

Kageura K, Umino B (1996) Methods of automatic term recognition – a review. Terminology 3(2):259–289

Liu H, Hu ZZ, Zhang J, Wu C (2006) BioThesaurus: a web-based thesaurus of protein and gene names. Bioinformatics 22(1):103–105

Okazaki N, Ananiadou S, Tsujii J (2010) Building a high quality sense inventory for improved abbreviation disambiguation. Bioinformatics 26(9):1246–1253

Thompson P, McNaught J, Montemagni S, Calzolari N, Del Gratta R, Lee V, Marchi S, Monachini M, Pezik P, Quochi V, Rupp C, Sasaki Y, Venturi G, Rebholz-Schuhmann D, Ananiadou S (2011) The BioLexicon: a large-scale terminological resource for biomedical text mining. BMC Bioinformatics 12:397

Wren JD, Garner HR (2002) Heuristics for identification of acronym-definition patterns within text: towards an automated construction of comprehensive acronym definition dictionaries. Methods Inf Med 41(5):426–434

Automatic Classification Methods

- ▶ Clustering Methods

Autonomy

Alvaro Moreno
Department of Logic and Philosophy of Science, University of the Basque Country, San Sebastian, Spain

Definition

The concept of autonomy expresses the property of a system that builds and actively maintains the rules that define itself, as well as the way it behaves in the world. So autonomy covers the main properties shown by any living system at the individual level: (1) self-construction (i.e., the fact that life is continuously building, through cellular *metabolisms*, the components which are directly responsible for the behavior of the system) and (2) functional action on and through the environment (i.e., the fact that organisms are *agents*, because they necessarily modify their boundary conditions in order to ensure their own maintenance as far-from-equilibrium, dissipative systems).

There are several key points. First, the notion of cyclic causal organization or recursivity remains the conceptual nucleus of autonomy as identity preservation. The system is constituted as a series of causal processes (energy transduction, component production, etc.) converging to a given (initial) state, as an indefinite repetition of the same loop (Bechtel 2007). Second, this identity is something more than a mere ordered pattern: It is a true set of mechanisms, whose function is the very maintenance of the system, and therefore of themselves. Third, this organization is intrinsically precarious because it depends on a dynamical order whose cohesion is dissipative. Fourth, an autonomous system is continuously performing actions, doing things that contribute to its own maintenance. These interactions with the environment are, at the same time, a result of the constitutive processes of the system and a necessary condition for their continuity. In other words, there is a reciprocal dependence between what defines the "self" (or the subject) and the actions derived from its existence, because it is not possible to separate the system's *doing* from its *being* (Moreno et al. 2008).

Characteristics

Although the idea of autonomy is commonly used, it is quite difficult to be defined. It is applied to very different objects, in rather different contexts, and with different rigor. Systems as diverse as machines, control devices, robots, computer programs, human beings, institutions, countries, animals, organs, cells, and others, apparently deserve to be called "autonomous," or at least to be studied and compared in terms of the "autonomy" displayed in their behavior. Originally the term was used in the context of law and sociology (in the sense of self-government of the Greek polis) or human cognition and rationality (in the sense of a cognitive agent that acts according to rationally self-generated rules).

Etymologically, autonomy means self-law (giving), namely, the capacity to act according to self-determined principles. Broadly speaking, autonomy is understood as the capacity to act according to self-determined principles. However, the idea of autonomy can adopt a more specific, minimal sense related to the capacity of a system to self-define, to construct its own identity. It is in this more basic sense that autonomy proves relevant for biology, to refer to the main feature of the organization of living organisms, namely, the fact that they are able to self-repair, self-produce, and self-maintain.

Although the idea of autonomy as a key concept to understand the organization of organisms has its roots in Kant (1790), it was the theory of autopoiesis which placed the notion of autonomy at the center of the biological understanding of living beings.

Within the autopoietic theory, the concept of autonomy has been applied to different biological domains: the basic metabolic domain, the immune domain, and the neural domain (Varela 1979). The root of the concept, however, was the characterization of the basic organization of the living, namely, its metabolic organization. Actually, it was a simplified and rather abstract version of a cyclic idea of metabolism that inspired Maturana and Varela (1973; Varela et al. 1974) to define living beings as autopoietic (self-producing) systems. Thus, the concept of autopoiesis refers to a recursive process of component production that builds up its own physical border. The global network of component relations establishes a self-maintaining dynamics, which brings about the

constitution of the system as an operational unit. In sum, components and processes are entangled in a cyclic, recursive production logic and they constitute an identity. According to Maturana and Varela, autonomous capacities stem from the logic of autopoiesis (Maturana and Varela 1973, 1980).

Although the stress is made in the circular logic of the system, (The idea that the essence of life consists in a cyclic or causally closed organization is also shared by other authors like Rosen (1981) or Ganti (1975).) in other places these authors also consider the relation of the system with its environment, admitting that autonomy implies the capacity to maintain the identity through the active compensation of deformations (Maturana and Varela 1980). Thus, phenomena like tornadoes, whirlpools, and candle flames, which are to a certain degree self-organizing and self-maintaining systems, are not autonomous, because they lack any form of active interaction exerted by the system. In that sense, what distinguishes simple forms of self-organization and self-maintenance from autonomy is that the former lack an internal organization which would be complex enough to be recruited for displaying selective actions, capable to actively ensure the maintenance of the system. (This is not to say that simple self-maintaining systems are necessarily less robust than autonomous ones).

However, Maturana and Varela have not analyzed which type of material organization may satisfy this requirement. Actually, their view on autonomy was explicitly abstract and functionalist. More recently, other authors have developed different approaches on the concept of biological autonomy. Hooker, Collier, Christensen, and Bickhard, for example (Christensen and Hooker 2000; Collier 1998; Bickhard 2000), have stressed both the material-energetic and the interactive dimension of autonomous systems, derived from the fact that the systems are in far-from-equilibrium conditions: since the cohesive organization of these systems exists only in far-from-equilibrium conditions, they must maintain an adequate interchange of matter and energy with their environment in order to keep this cohesion. Otherwise these systems will disintegrate. Therefore, an autonomous system needs an internal mechanism able to organize and channel energy flows for the system's self-maintenance.

A similar concern in formulating the idea of autonomy in material terms is probably on the basis of Kauffman's (2000) approach. Starting from the concept of an autocatalytic set, the main condition required to consider a system as autonomous is that it be capable of performing what this author calls (at least) one "work-constraint cycle." This capacity is implemented through a deep entanglement between work and constraints: "work begets constraints begets work," as some form of closure in the abstract space of catalytic tasks. This idea is based, on the one hand, in Atkins' view (1984) of work as a coordinated, coherent, and constrained release of energy, and, on the other hand, in the recognition that work is absolutely necessary to build constraints. To be autonomous, a system must do work; to be capable of work it requires constraints to channel the flow of energy in an appropriate way; and to build those constraints the system requires appropriately constrained energy flows, that is to say, work. This is the circularity of the "work-constraint cycle." Kauffman's account envisages how functionality or utility (implicit in the idea of work) can come out of the causal circularity of the system, where this circularity is not only understood in terms of abstract relations of component production, but also as a thermodynamic logic sustaining the specific chemical recursivity of the system: All the processes ongoing in the system are constrained in order to satisfy the condition of self-maintenance.

Implications

The idea of autonomy can serve as a bridge from the nonliving to the living domain (Ruiz-Mirazo et al. 2004). Autonomy is applicable to this gap between chemistry and biology, beyond standard self-organization, because in a living organism component production relationships (chemical transformations, reaction feedbacks, mutual interactions, catalytic effects, etc.) can be interpreted as *self-constraining* processes, that is, as the generation by the very organism of the local rules (constraints) that govern its dynamic behavior (Pattee 1972). It is through this self-constraining action that the organism actually defines itself, *constructs an identity of its own*.

The concept of autonomy has been proposed for defining the very nature of biological individuals, because it is intended to account for the complex material organization underlying any living organism,

namely, its *metabolism, understood as* a cyclic, self-maintaining network of reactions by means of which the components of an organism are continually produced.

Cross-References

- ▶ Closure, Causal
- ▶ Constraint
- ▶ Function
- ▶ Organization
- ▶ Perturbation
- ▶ Robustness
- ▶ Self-Organization
- ▶ Systems, Autopoietic

References

Atkins PW (1984) The second law. Freeman, New York
Bechtel W (2007) Biological mechanisms: organized to maintain autonomy. In: Boogerd F, Bruggeman F, Hofmeyr JH, Westerhoff HV (eds) Systems biology: philosophical foundations. Elsevier, Amsterdam, pp 269–302
Bickhard M (2000) Autonomy, function, and representation. Communication and Cognition - Artificial Intelligence 17(3–4):111–131
Christensen W, Hooker C (2000) Anticipation in autonomous systems: foundations for a theory of embodied agents. Int J Comput Anticip Syst 5:135–154
Collier J (1998) Autonomy in anticipatory systems: significance for functionality, intentionality and meaning. In: Dubois D (ed) Proceedings of CASYS'98, the second international conference on computing anticipatory systems. Springer-Verlag, New York
Kant I (1790:1952) Critique of judgement. Oxford University Press, Oxford
Kauffman S (2000) Investigations. Oxford University Press, Oxford
Maturana H, Varela F (1973) De máquinas y Seres Vivos. Autopoiesis: La organización de lo Vivo. Editorial Universitaria. Santiago (1994, 3rd edition including new prefaces by each author)
Maturana H, Varela F (1980) Autopoiesis and cognition: the realization of the living. Reidel, Dordecht
Moreno A, Etxeberria A, Umerez J (2008) The autonomy of biological individuals and artificial models. Biosystems 91(2):309–319
Pattee H (1972) Laws and constraints, symbols and languages. In: Waddington CH (ed) Towards a theoretical biology, vol 4, Essays. Edinburgh University Press, Edinburgh, pp 248–258
Rosen R (1991) Life itself: a comprehensive inquiry into the nature, origin and fabrication of life. Columbia University Press, New York
Ruiz-Mirazo K, Moreno A (2004) Basic autonomy as a fundamental step in the synthesis of life. Artif Life 10(3):235–259
Ruiz-Mirazo K, Pereto J, Moreno A (2004) A universal definition of life: autonomy and open-ended evolution. Orig Life Evol Biosph 34:323–346
Varela F (1979) Principles of biological autonomy. Elsevier North Holland, New York
Varela F, Maturana H, Uribe R (1974) Autopoiesis: the organization of living systems, its characterization and a model. Biosystems 5:187–196

Average Path Length

- ▶ Characteristic Path Length

Avidian

- ▶ Digital Organism

Azacitidine

- ▶ 5-azaCytosine