A Computational Theory of Meaning

Pieter Adriaans

SNE group, IvI
University of Amsterdam,
Science Park 107
1098 XG Amsterdam,
The Netherlands.

Abstract. In this paper we propose a Fregean theory of computational semantics that unifies various approaches to the analysis of the concept of information: a meaning of an object is a routine that computes it. We specifically address the tension between quantitative conceptions of information, such as Shannon Information (Stochastic) and Kolmogorov Complexity (Deterministic) and the notion of semantic information. We show that our theory is compatible with Floridi's veridical theory of semantic information. We address the tension between two-part codes and Kolmogorov complexity, which is a form of one-part code optimization. We show that two-part codes can be interpreted as mixed models that capture both structural and stochastic aspects of the underlying data set. Two-part codes with a length close to the optimal code indeed exhaust the stochastic qualities of the data set, as Vitányi and Vereshchagin have proved, but it is not true that this necessarily leads to optimal models. This is due to an effect that we call polysemy, i.e. the fact that under two-part code optimization a data set can have different optimal models, with no mutual information. This observation destroys the possibility to define a general measure of the amount of meaningful information in a data set based on two-part code compression, despite many proposals in this direction (MDL, Sophistication, Effective Information, Meaningful Information, Facticity etc.). These proposals have value, but only in the proper empirical context: when we collect a set of observations about a system or systems under various conditions, data compression can help us to construct a model identifying the invariant aspects of the observations. The justification for this methodology is empirical, not mathematical.

keywords: two-part code optimization, Kolmogorov complexity, Shannon information, meaningful information

1 Introduction

This paper describes some results in the context of a long term research project that aims to understand learning as a form of data compression [34]. There is an abundance of practical applications of learning by compression: well known machine learning algorithms like decision tree induction, support vector machines, neural networks and many others can be interpreted as processes that incrementally exploit regularities in the data set to construct a model [37], classification and clustering algorithms based on standard industrial compression software perform surprisingly well in many domains [22; 35] and data compression seems to play an essential role in human cognition [21].

2 Meaning as computation

The idea that an intension of an object could be a computation was originally formulated by Frege [1]. The expressions "1 + 4" and "2 + 3" have the same extension (Bedeutung) "5", but a different intension (Sinn). There are contexts in which such a distinction is necessary. Consider the sentence "John knows that $\log_2 2^2 = 2$". Clearly the fact that $\log_2 2^2$ represents a specific computation is relevant here. The sentence "John knows that $2 = 2$" seems to have a different meaning. In this sense one mathematical object can have an infinity of different meanings and by necessity only a finite fraction of these possibly compress the object.

Frege's theory of meaning can be interpreted in the context of a general theory of computing:

Definition 1. *Given the general model of computation $program(input) = output$ we say that for an instance of the model $p(x) = y$, the expression $p(x)$* denotes *the object y and is a semantics for it.*

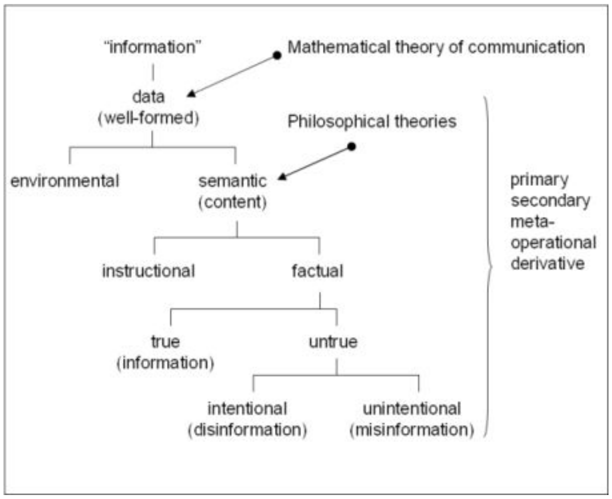

Fig. 1. Floridi's classification of information

In short,the phrase $p(x)$ is a computation that specifies a meaning for y. A computational semantics is an operational semantics: it tells us what we need to *do* to transform one data set into another. If we analyze the expression $p(x) = y$ using Floridi's classifitation (See figure 1), then we can observe that p, x and y are well-formed strings, which makes $p(x) = y$ a well-formed expression. The strings x and y are simple well-formed data, while the program p is a form of instructional information. The sentence $p(x) = y$ is true if the instructions specified in p performed on x produce y. Consequently the sentence $p(x) = y$, since it consists of data that are well-formed and meaningful, satisfies Floridi's definition of strong semantic information,when it is true. A Fregean theory of meaningful information allows us to unify philosophical approaches to the analysis of meaningful information (Bar-Hillel [5], Floridi [39]) with the computational approach to complexity theory. In the context of such a theory of computational semantics a number of philosophical issues have a clear interpretation. We shortly discuss two instances:

1. Floridi's fourth criterion of truthfulness has been much debated in recent literature: can it really be that untrue sentences do not contain any information [45]. The association with computation allows us to understand this issue better. The phrase $p(x)$ specifies a meaning, but until we know that the computation actually stops we cannot say that it gives us information. Given the halting problem there is no way to decide this question without running the program $p(x)$. So we can say the expression $p(x) = y$ has an objective meaning $p(x)$ but that it only contains semantic information when it is true. It is easy to write a simple program that will find *the first number that violates the Goldbach's conjecture*, but when the conjecture is true, such a program will never stop. In this case the expression has a meaning, but no denotation.
2. There is an infinite number of different computations that generate the same data set, consequently a data set has an infinite amount of different meanings. Kolmogorov complexity asymptotically measures the amount of meaningful information in a data set by measuring the length of the shortest program (i.e. meaning) that generates the set. Since there may be various programs that give an optimal compression of the data set, there is not one unique intrinsic meaning for a data set. Still we can measure the amount of information in a meaning $p(x)$ even if the program has no accepting computation, i.e. no denotation.

Observation 1 *In a Fregean theory of meaning we can objectively quantify the amount of meaning in a data set, even if the meaningful information is not truthful, but we cannot specify one intrinsic meaning for a data set.*

In essence the conception of meaning is polyvalent, complex and subjective. A poem or a painting, a fragment of a song or even a blob of paint on the wall may be meaningful for one person and not for another. In its crude form the concept of meaning seems to be purely subjective: any object can have any meaning for any subject.

In this context a simple object might have more meanings than the actual information it contains and compression does not help us to reconstruct it. This changes as soon as we collect observations in a systematic way. From a series of observations we can in some cases construct a real general model. We get a version of two-part code when we interpret the program p in the expression $p(x) = y$ as a predicate or model for y. The rationale behind this proposal is that in many cases, when $p(x) = y$ is true the program p contains information about the data set y.

Note that there are many cases in which p cannot be said to be a model of y, e.g. consider $y = x$ and p is the print program. After execution of the program we know that it is true that y has been printed, which is semantic information but does not tell us anything about the content of y. At the other extreme, suppose that p is a program the generates Fibonacci numbers and x is an index then we can say that p really is a model of y as the x-th Fibonacci number.

3 Models, theories and knowledge

We learn from experience. Living beings interact with the world and they benefit from its regularities. It is natural that they should have a general capacity to observe these regularities and construct models that allow them to predict future states of the world and adapt their behavior accordingly.

Historically the notions of true knowlegde, a good model and the right definition are intricately related. Plato's theory of ideas implies that there are intrinsic models, that allow us to know all the relevant structural aspects of imperfect beings in our world. There is an idea of 'horseness' that describes all the relevant aspects of the horses around us. These ideas, also known as universals (in modern jargon we would say models), are perfect and belong to a world beyond space and time. Knowing the right definition of an object gives us true knowledge about the object.

Aristotle and others rejected the theory of ideas. Some philosophers in the middle ages even defended the theory that the predicate 'man' was a mere name, without any ontological implications. This position, known as 'nominalism', was defended, amongst others, by Occam (1288-1347). The central motivation is a principle of parsimony: why introduce a world of ideas if you do not really need it to do science? The accepted formulation in Latin is “Pluralitas non est ponenda sine necessitate”, i.e., “Plurality is not to be posited without necessity”. This rule (that is not explicitely stated in the writings of Occam) became famous as Occam's razor, because it helps us to cut off the 'beard of Plato', i.e. his superfluous notion of universals. In modern science Occam's razor still plays a role, not in an ontological sense, but as a heuristic principle: simple theories are to be preferred above more complex ones.

In current scientific methodology the sequential aspects of the learning process are formalized in terms of the empirical cycle, which according to de Groot [8] has the following stages:

1. Observation: The observation of a phenomenon and inquiry concerning its causes.
2. Induction: The formulation of hypotheses - generalized explanations for the phenomenon.
3. Deduction: The formulation of experiments that will test the hypotheses (i.e. confirm them if true, refute them if false).
4. Testing: The procedures by which the hypotheses are tested and data are collected.
5. Evaluation: The interpretation of the data and the formulation of a theory - an abductive argument that presents the results of the experiment as the most reasonable explanation for the phenomenon.

Thus learning is a sequential process and it analyzes sequential processes. In the context of information theory the set of observations will be a data set and we can construct models by observing regularities in this data set.

3.1 Optimal models and data compression

We distinguish deterministic processes and random or stochastic processes. For the former the future will be exactly the same when we restart the process under the same conditions, for the latter this is not the case. We may model stochastic processes by means of a probability distribution and deterministic processes by means of a program on a deterministic computer. Consequently the regularities we find in a data set can be of a stochastic nature: certain observations have a higher frequency than others, or deterministic: after event A we always observe event B. There are two corresponding ways to measure information:

- From a deterministic perspective: $K(M)$ is the Kolmorgorov complexity of a data set M, seen as a set of messages or observed events. It is the length of the smallest deterministic computer program that generates the data.
- From a stochastic perspective: The Shannon information $I(m) = -\log p(m)$, relates the amount of information m in a single message to the probability $p(m)$ of that message or event occurring.

Shannon information formalizes a notion that has been known at least since Samuel Morse (1791-1872) designed his famous code: it is more efficient to assign shorter codes to more frequent symbols, especially when resources are scarce. Morse assigned only 1 unit for high frequency letters like e and t and 4 units for low frequency letters like y and q. Shannon information links the frequency to optimal code for a message type and provides a basis for data compression. Kolmogorov complexity allows us to look into the internal structure of the data set. With the theory of Shannon the best we can do when we observe a data set like 0101010101010101010101010101010101010101 is assign a uniform distribution to the messages $p(0) = p(1) = 0.5$. Consequently its optimal Shannon code is 40 bits long. But such a string is with high probability too regular to be stochastic. Kolmogorov complexity can deal with the internal structure of this message, that is apparently from a deterministic source. It will model the data set as $For\ i = 1\ to\ 20\ PRINT(‘01')$. The relation between frequency, probability, optimal code and data compression suggests that information theory might be a tool for modeling learning processes. Indeed many learning algorithms, like C4.5 and Support Vector Machines, already deploy some form of information theory. In this paper we study learning from a general information theoretical perspective. The central research question is:

Question 1. Given a set of observations represented in a data set, can we use information theory to construct the right model, i.e. the model that gives the best predictions.

For the two extremes we have relatively good theories. If the process we observe is purely stochastic then its optimal model or sufficient statistic is a probability distribution of the set of possible events. According to Shannon's information theory the data set can be recoded and the optimal code, which gives the maximal compression, is dependent on the entropy of the set of messages. If the data set is big enough we can easily estimate the probability distribution that defines the optimal compression.

If the process is deterministic we can model it as a program on a universal computer. Its optimal code for a given universal machine will be the shortest program that produces the data set. The length of shortest program is its Kolmogorov complexity, which also defines a universal a priori probability for the data set. It is not possible to compute such a program, but this should not keep us from using the theory on existing data sets. If we can compress a data set, this is significant information. The consequences of the non-computability of Kolmogorov complexity have been analyzed in [42]: in everyday life the possibility that we miss models for compressible data sets, because of the uncomputability of Kolmogorov complexity is neglectable. From this point of view Kolmogorov complexity is a reliable theory to study learning as data compression.

3.2 Mixed models

Purely deterministic or stochastic processes are rare in the every day world around us. Most phenomena have a mixed nature: some aspects are deterministic some are random. For such processes learning involves the separation of random aspects from deterministic ones. Consequently we need mixed models: partly deterministic, partly stochastic. For such mixed processes the situation is less clear. If we combine Shannon's notion of an optimal code with Bayes' law, we get a rough theory about optimal mixed model selection. Let $\mathcal{H}$ be a set of hypotheses and let x be a data set. Using Bayes' law, the optimal computational model under this distribution would be:

$$M_{map}(x) = argmax_{M \in \mathcal{H}} \frac{P(M)P(x|M)}{P(x)} \tag{1}$$

This is equivalent to optimizing:

$$argmin_{M \in \mathcal{H}} - \log P(M) - \log P(x|M) \tag{2}$$

Here $-\log P(M)$ can be interpreted as the length of the optimal *model code* in Shannon's sense and $-\log P(x|M)$ as the length of the optimal *data-to-model code*; i.e. the data interpreted with help of the model. This insight is canonized in the so called:

Definition 2. *Minimum Description Length (MDL) Principle : The best theory to explain a data set is the one that minimizes the sum in bits of a description of the theory (model code) and of the data set encoded with the theory (the data to model code).*

The MDL principle is often referred to as a modern version of Occam's razor [44], although in its original form Occam's razor is an ontological principle and has little to do with data compression. In many cases MDL is a valid heuristic tool and the mathematical properties of the theory have been studied extensively [32]. Still MDL, Occam's razor and two-part code optimization have been the subject of considerable debate in the past decennia (e.g. [24]). MDL defines the concept of 'the best theory', but it does not explain why the definition should work. Historically it is ironic that the name of Occam should be associated with the notion of the 'best' theory, since the idea that there exists such a thing as an intrinsic model, is close to the ideas of Plato, a position that Occam attacks. We might interpret the MDL principle in a weak nominalistic way as a heuristic principle that simply favors small theories or we might give a strong Platonic reading: balanced data compression techniques really give us the intrinsic best theory, with the best generalization and prediction. In the latter case we associate ourselves with a position that might be called:

Definition 3. *Information Theoretical Platonism (ITP): i.e. the theory that for every data set there is one intrinsic model and that this model can be found by means of data compression techniques.*

If such an intrinsic model exists, we can measure its complexity, consequently there is a plethora of proposals for measures of meaningful or useful information related to this notion of Platonism:

- Esthetic Measure (Birkhoff, Bense [7], [3], [15]).
- Sophistication (Koppel, [10], [25], [29])
- Logical Depth (Bennet, [11])
- Statistical complexity (Crutchfield, Young , [13], [14], [16])
- Effective complexity (Gell-Mann and Lloyd, [23])
- Meaningful Information (Vitányi, [28])
- Self-dissimilarity (Wolpert and McReady, [31])
- Computational Depth (Antunes et al., [29])
- Facticity (Adriaans [40])

The fact that none of these proposals has found general acceptance is a sign that there might be something wrong with the basic intuitions behind ITP. Domingos has argued against the validity of Occam's razor in machine learning:

the smallest model is not necessarily the best [19]. Domingos explicitly targets MDL in his analysis: *Information theory, whose goal is the efficient use of a transmission channel, has no direct bearing on KDD, whose goal is to infer predictive and comprehensible models from data. Having assigned a prior probability to each model in the space under consideration, we can always recode all the models such that the most probable ones are represented by the shortest bit strings. However, this does not make them more predictive, and is unlikely to make them more comprehensible.* Domingos makes a valid point about comprehensibility, which is not an ambition of two-part code optimization.

His objections to the selection of short models however are misguided for various reasons: 1) two-part code does not select the shortest model but a model that optimizes the balance between a compressed model of the theory and the data coded, given the theory, i.e. it selects a theory that is short but not too short and 2) the probability assigned to the model under Kolmogorov theory is related to the length of its optimal code and as such it is an objective asymptotic measure, not some ad-hoc choice. Thus the claim for a general theory of induction (which is implied by ITP), that has been made by some authors ([18; 38]), is stronger than the objections suggest.

3.3 Polysemy: optimal models with no mutual information

But there are other problems. In [36] we already proved that incremental compression does not necessarily produces good theories. An attempt to formulate a definitive theory of meaningful information based on data compression is [40], but already in this paper we were forced to conclude that in the limit, for very large data sets, universal Turing machines would be selected as optimal models, which is counter intuitive. In the meantime other doubts about the validity of this approach have been raised [43]. The idea behind two-part code optimization as a method for selecting the best model also seems to be at variance with the no free lunch theorems [27]: "*(...) any two optimization algorithms are equivalent when their performance is averaged across all possible problems"*. Yet, the reasoning behind the results in [20] and related proposals for two-part code optimization is clear: if we can find a two-part code model for a data set that is 'close' to its optimal code then both the model-code and the data-to model code must be incompressible, therefore random and thus by necessity the model must contain an exhaustive description of all relevant stochastic qualities of the data set. Consequently it must be the 'best' model. Thus we have two theories: one that gives us, in a sense, the best model and one that tells us, in a sense, there is no best model.

In this paper we prove that the two views can be reconciled, at least partly, by observing the fact that two-part code optimization is polysemantic: i.e. a data set can have several different optimal models that have no mutual information, but that all compress the set close to its Kolmogorov complexity. Polysemy is already implied by the notion of invariance that is central to Komogorov complexity: Several different algorithms may give an optimal compression, but we can be sure that the length of these programs always vary within a constant.

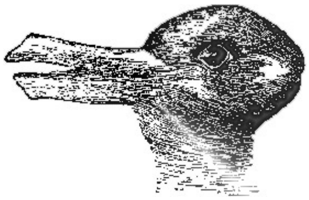

Fig. 2. Wittgenstein's Duck-Rabbit

The concept of polysemy is important in many domains besides machine learning. It plays a role in empirical sciences, it is the basis for art as imitation and it is an important feature of human cognition. In psychology the concept of polysemy is known as a gestalt switch: a shift in interpretation of a data set where structural data and ad hoc data switch roles. Consider the image used by Wittgenstein in the Philosophical Investigations of the rabbit that, when turned 90 degrees, appears to be a duck. One could say that, from a cognitive point of view, the predicate '*rabbit*' is an optimal description of the image, as well as '*duck*', but that the description as '*image of a rabbit that appears to be a duck when turned 90 degrees*', is too rich and does not help to compress the image as well as the shorter descriptions. In this case a description that gives all the structural aspects of the image is not the best description to compress it. The image really has two, mutually exclusive, meanings, although the meanings have mutual elements e.g. the eye. The ad hoc arrangement of the rabbits ears is essential for the structural description as a duck's beak, while the ad hoc irregularities at the back of the head of the duck are vital for its interpretation as a rabbit's mouth.

Obviously, since the models have restricted mutual information, their capacity for generalization (duck versus rabbit) and prediction (flying versus jumping) varies. What is more, if a data set of at least two optimal models with no or little mutual information, it will also have a spectrum of related nearly optimal models that share information with both of them. All of these models will have different capacity for generalization. This leads to the conclusion that the strong Information Theoretical Platonism (ITP) interpretation of two-part code optimization is wrong: there may be more than one best model with fundamentally different generalizations. On the other hand data compression is an important heuristic principle: if a data set is uncompressible there are no regularities to be observed and thus no predicitive models can be constructed.

Observation 2 *Data compression is a necessary but not a sufficient condition for selecting good models.*

This does not imply that every compression leads to useful models. Two-part code optimization does tell us, that when we are confronted with two different theories that explain a data set, we should choose the one that compresses the data set better. Still, even in this weak reading is it prima facie not clear why this should be a good directive.

3.4 Empirical justification for two-part code optimization

A justification for two-part code optimization emerges if we study it in the context of empirical science. The empirical method forces us to generalize the notion of meaning in two dimensions: 1) invariance over observers and 2) invariance over time. There is a connection between these notions of invariance and data compression. If one collects various observations over time of the same phenomenon by various observers in a data set, general invariant patterns will emerge that can be separated from all kinds of ad hoc aspects of the observations. The invariant patterns will allow us to compress the data set. In this conception of empirical science there is a real relation between compression of data and the construction of useful models. Note that the fact that generalization over observers in time gives us reliable models is itself empirical. There is no a priori mathematical motivation for it:

Observation 3 *The justification for two-part code optimization is ultimately empirical and not mathematical.*

With these observations in mind we can adapt the definition 2 of MDL to one that better fits the empirical aspects.:

Definition 4. *Empirical Minimum Description Length (EMDL) Principle :*

If we select a data base of a sufficient number of descriptions of observations of the same phenomenon under different circumstances then, with high probability, the theory that minimizes the sum in bits of a description of the theory (model code) and of the data set encoded with the theory (the data to model code), gives the best description of the invariant aspects of the phenomenon and thus gives the best generalized model with the best predictive capacities.

MDL works as a consequence of the fact that the empirical method world works. The empirical method itself is designed to separate ad hoc phenomena from structural ones. Applied to the Wittgenstein gestalt switch example the polysemy would disappear if we interpreted the picture in the context of a series of observation of ducks or rabbits.

4 Information theory and two-part code optimization

4.1 A spectrum of entropy and information measures

The unreasonable effectiveness of mathematics in information theory can partly be explained by the dual nature of logarithmic operations: they can be used to

General context
A discretized system V of size l with k possible micro states.
Total number of possible configurations k^l

Elementary characterization of Entropy via the logarithmic operation
Hartley entropy: $H_0(A) = \log_b |A| = \log_b k^l$ i.e. the length of an index of an element of the set of possible states. The equivalent in Physics is in Physics Boltzmann entropy: $S_B = k \ln W$

Generalizations

Deterministic Systems	**Various proposals for Mixed Systems**	**Stochastic Systems**
Information theory		
Kolmogorov complexity: The length of the index of the smallest machine that generates/describes the system: $K(V) = K_U(S\mid\epsilon) = \min_i\{l(\bar{i})\mid U(\bar{i}\epsilon) = V\}$	Two part models: Minimum Description Length (MDL): $argmin_{M\in\mathcal{H}}$ $-\log P(M) - \log P(x\mid M)$ Kolmogorov's structure function: $h_x(\alpha) = \min_S\{log\lvert S\rvert : x \in S, K(S) \leq \alpha\}$ Non-deterministic Turing machines	Shannon Entropy: The length of an index optimized for probability/density: $l(S(V))$ where $S(V) = -\Sigma_i p_i \log p_i$
Physics		
Classical Mechanics Descriptive complexity of a mechanical system at $t = 0$.	Long distance correlations: Tsallis Entropy $S_q(p_i) = \frac{k}{q-1}(1 - \Sigma_i {p_i}^q)$, Rényi entropy: $H_\alpha(V) = \frac{1}{1-\alpha}\log(\Sigma_{i=1}^n p_i^\alpha)$ Quantum information: $\lvert\psi\rangle = \alpha\lvert 0\rangle + \beta\lvert 1\rangle$	Thermodynamics Gibbs entropy $S_G(V) = -k\Sigma_i p_i \ln p_i$

Table 1. An overview of various notions of quantitative notions of information and entropy and their interrelations.

express, both, our uncertainty when making a choice from the elements of a set A, as well as, the length of a discrete symbolic representation of an element given the set A. On top of that we can claim universality for the log operation. As such it mediates between the Platonic world of Natural and Real numbers and the outside world of manipulation of discrete physical objects.

There is a spectrum of information and entropy measures that capture different extensional aspects of systems (See table 1). Suppose we have a discretized system S of a certain size l whose individual elements can be in a finite number of different states k. We have a number of ways to characterize the system without loss of information. The most concise one is the Hartley entropy $H_0(A) = \log_b |A| = \log_b k^l$, which has two interpretations:

- Probability: As a quantification of our lack of knowledge before we choose an element of A: a characterization of the amount of information we get if we pick a sample from finite set A, uniformly at random. This notion is stochastic, non-deterministic and has representation in real numbers $\mathbb{R}$.
- Representation: As a quantification of the amount of knowledge we have, when we know a typical element of A. It corresponds to the length of an index of an element of A in the set of the total number of possible states of the total system $H_0(A) = \lceil \log_b |A| \rceil = \lceil \log_b k^l \rceil$. This notion is symbolic, deterministic and has a representation in discrete symbols and natural numbers $\mathbb{N}$.

The length of such an index $\log_a k^l$, where a is the unit of measurement, characterizes the complexity or entropy of the system. For $a = 2$ we get bits, for $a = e$ we get gnats, but any other base for the log operation is possible. From a philosophical point of view it is important to notice that the Hartley function is not a definition but that it corresponds to a theorem in number theory: it can be proved that the Hartley entropy, measured in bits is the only mathematical function on the set of natural numbers $\mathbb{N}$ that satisfies a) additivity $H(mn) = H(m) + H(n)$, monotonicity $H(M) \leq H(m+1)$, normalization $H(2) = 1$. This implies that when we want to have a quantitative measure in bits for the amount of information in sets that is additive and monotone, the Hartley function is the only option we have:

Observation 4 *The logarithm gives a universal characterization of additivity of multiplication [4].*

In physics this corresponds to the Boltzmann entropy $S_B = k \ln W$, where W is the number of states of the system. We use the natural logarithm and the Boltzmann constant k as normalization factor.

The entropy represented as logarithm can be generalized in three directions:

- Stochastic: A generalization to non-uniform distributions. In representational term one could interpreted this in terms of an index optimized for density, where the length of code $\log_a(\frac{\#k_i}{l})$ for an element in state k_i is dependent on its relative frequency $\frac{\#k_i}{l}$. Such a characterization assumes that the states of the individual elements are independent of each other. This notion of entropy is associated with Shannon information and the Shannon entropy gives the expected length of the code $lS(V)$ where $S(V) = -\Sigma_i p_i \log p_i$. The corresponding domain in physics is thermodynamics with Gibbs entropy $S_G(V) = -k\Sigma_i p_i \ln p_i$.

- Symbolic: As the index/representation of a deterministic computer program p that generates S on a universal symbolic computer. There will be a shortest program. Its length characterizes the information content of the data set. This characterization assumes that the individual elements are completely determined and thus interdependent. Only deterministic processes generate such data sets. The corresponding information measure is Kolmogorov complexity. The corresponding domain in physics is classical mechanics.
- Various Mixed models that combine derterministic and stochastic elements. In information theory this takes the form of proposals for mixed models that combine a structural computational part with an ad hoc stochastic part. Both MDL and Kolmogorov's structure function have this form. We can also develop a more general model by using non-deterministic machines as a more general model of computation but that is beyond the scope of this paper. In physics there are proposals for variants of entropy that are sub-additive and can model long distance correlations in the form of Rényi and Tsallis entropy. Quantum bits are also a generalization of the classical information concept in this sense.

4.2 Shannon information and Kolmogorov complexity

There are many correlations between the central concepts of the theory of Kolmogorov complexity and Shannon information. An overview is given in table 2. For stochastic sequences of messages the Shannon optimal code length and the Kolmogorov complexity approximate each other asymptotically.

We follow the standard reference for Kolmogorov complexity [33]. When we select a reference universal Turing machine U from $\mathcal{U}$ we can define the prefix-free Kolmogorov complexity $K(x)$ of an element $x \in \{0,1\}^*$ the length $l(p)$ of the smallest program p that produces x on U. We first define the conditional complexity:

Definition 5. $K_U(x|y) = \min_i\{l(\bar{i}) : U(\bar{i}y) = x\}$

The actual Kolmogorov complexity of a string is defined as the one-part code:

Definition 6. $K(x) = K(x|\varepsilon)$

Here all the compressed information, model as well as noise, is forced on to the model part.

Definition 7. *The randomness deficiency of a string x is $\delta(x) = l(x) - K(x)$*

For two universal Turing machines U_i and U_j, satisfying the invariance theorem, the complexities assigned to a string x will never differ more than a constant:

$$|K_{U_i}(x) - K_{U_j}(x)| \leq c_{U_i U_j} \tag{3}$$

The invariance theorem also holds for the randomness deficiency of a string:

Lemma 1. $|\delta_{U_i}(x) - \delta_{U_j}(x)| \leq c_{U_i U_j}$

Proof: $|\delta_{U_i}(x) - \delta_{U_j}(x)| = |(l(x) - K_{U_i}(x)) - (l(x) - K_{U_j}(x))| = |K_{U_i}(x)) - K_{U_j}(x))| \leq c_{U_i U_j}$□

Note that the fact that the theory of computing is universal is of central importance to the proof of the invariance theorem: the proof deploys the fact that any machine can emulate any other machine. A string is Kolmogorov random with reference to a universal Turing machine if there is no program that compresses it. Using randomness deficiency the definition is:

Definition 8 (Kolmogorov randomness).
$Random_U(x)$ iff $\delta_U(x) \leq c_r$

Usually the constant c_r is taken to be 0 in this definition, but $\log l(x)$ is better (See Appendix 7). Note that *Random*(...) is a meta-predicate.

A choice for a universal reference Turing machine U generates a specific so-called 'universal distribution' m_U over the set of strings:

$$m(x) = \Sigma_{p:U(p)=x} 2^{l(p)} \tag{4}$$

The relation with information is clear from Levin's coding theorem: $\log m_U(x) = K_U(x) + O(1)$

Shannon information and Kolmogorov complexity are complementary concepts. The strength of the one is the weakness of the other. If we analyze deterministic strings with Shannon's technique we may very well assign maximum entropy to a highly compressible string. Conversely, when we analyze a stochastic data set with Komogorov complexity we have to store the definition of the optimal code in the program that compresses the data set. For smaller sets this gives a large overhead up to the point that Shannon compressible data sets are considered to be random by Kolmogorov complexity. Since Shannon's theory is not part of the definition of universal computing Kolmogorov complexity has to 'rediscover' the concept for each stochastic data set it tries to compress.

Although there are many open questions concerning the exact relation between Shannon information and Komogorov complexity table 2 illustrates that the two concepts are interrelated and seem to be different approximations of the same reality. Shannon information starts from probability and defines optimal code. Kolmogorov complexity starts from optimal code and defines probability. There exists a plethora of proposals for mixed models of entropy that generalize aspects of both approaches.

4.3 Mixed entropy models and two-part code optimization

In the context of Kolmogorov complexity it is natural to interpret fomula 2 as (See e.g. [28]):

Kolmogorov Complexity	**Shannon information**
Deterministic	Stochastic
Based on Computation	Based on Probability
Uncomputable	Computable
Single message: $V \in \{0,1\}^*$	Sequence of messages: $V = (x_1, x_2, , x_n)$
Descriptive Complexity of a data set: $K_U(V) = \min_i\{l(\bar{i}) : U(\bar{i}) = V\}$	Communication Entropy of a data set: $S(V) = -\Sigma_{I \in S} p_i \log_2 p_i$
Randomness deficiency: $\delta(V) = l(V) - K(V)$ Where $l(V)$ is the length of data set	Absolute redundancy: $D = R - r$ $\lvert V \rvert$ is the cardinality of the sequence. The absolute Rate is: $R = \log \lvert V \rvert$. The rate r is the average entropy per message
Probability is defined on the basis of the length of code $m(V) = \Sigma_{p:U(p)=V} 2^{l(p)}$	Length of optimal code is defined on the basis op probability: $-\log P(x)$
Information in a single data set: $-\log m(V) = K(V) + O(1)$	Information in a single message: $I(x) = -\log_2 P(x)$.
A string is random if and only if it is incompressible and has zero randomness deficiency.	A memoryless source with a uniform distribution has zero redundancy (and thus 100% efficiency), and cannot be compressed.

Table 2. A comparison of central concepts of the theory of Kolmogorov complexity and Shannon information.

$$argmin_{M \in \mathcal{H}} K(M) + K(x|M) \tag{5}$$

In this way one obtains, within the limits of the asymptotic accuracy of Kolmogorov complexity a theory of optimal model selection. A similar proposal that literally combines computational and stochastic elements is the structure function proposed by Kolmogorov in 1973 [20]. The structure function specifies a two-part description of data set, 1) the regular part: a predicate that describes a set of strings, measured in terms of its Kolmogoroc complexity $K(S)$ and 2) an ad-hoc part: the index of the data set in this set measured as the corresponding Hartley entropy $\log |S|$.

$$h_x(\alpha) = \min_S \{log|S| : x \in S, K(S) \leq \alpha\} \tag{6}$$

Vereshchagin and Vitányi prove a very strong result [20]. The model selected by the structure function is optimal: "*We show that the structure function determines all stochastic properties of the data: for every constrained model class it determines the individual best fitting model in the class irrespective of whether the true model is in the model class considered or not.*"

The basis for the optimality claim is the fact that the set S defines an algorithmic sufficient statistic:

Definition 9. *The set S defines an* algorithmic sufficient statistic *for an objectx if:*

- *S is a* model*, i.e. we have $K(x|S) = \log |S| + O(1)$, which gives a constant* randomness deficiency $\log |S| - K(x|S) = O(1)$*. This implies that x is typical in S, i.e. S defines all statistical relevant aspects of x.*
- *We have* information symmetry $K(x, S) = K(x) + O(1)$*, i.e. S should not be too rich in information, but specifiy just enough to identify x.*

This makes the structure function prima facie a sort of holy grail of learning theory. It suggests that compressible data sets have an inherent 'meaning', i.e. the optimal predicate that describes their essential structure. It also gives us an interpretation of the notion of randomness: a data set x is random iff it only has the singleton set $\{x\}$ as a model. Based on such observations some authors have claimed Komogorov complexity to be a "universal solution to the problem of induction" [18; 38]. Unfortunately attempts to develop a theory of model selection and meaningful information based on these results have not been very successful [23; 24; 28; 31; 40; 43].

4.4 Two-part code optimization and polysemy

The two views can be reconciled by realizing that two-part code optimization is polysemantic:

Definition 10. *A computational system is (strongly) polysemantic under two-part code optimization if it has data sets that within the variance of a constant c have at least two optimal models that have no mutual information.*

Polysemy is a natural condition for Turing equivalent systems:

Theorem 1. *The class of Turing machines is polysemantic under two-part code optimization.*

Proof: this is a direct consequence of the invariance theorem itself. Suppose $U1$ and $U2$ are two universal Turing machines and x is a data set. We select $U1$ and define the standard complexity: $K_{U1}(x|y) = \min_i\{l(\bar{i}) : U1(\bar{i}y) = x\}$ with $K_{U1}(x) = K_{U1}(x|\varepsilon)$. Suppose we interpret $\bar{i}y$ as a two part code with i the index of a machine and y its input. We can also select $U2$ as our reference machine. We get: $K_{U2}(x|y) = \min_i\{l(\bar{j}) : U2(\bar{j}z) = x\}$ and $K_{U2}(x) = K_{U2}(x|\varepsilon)$. We know that $K_{U1}(x) = K_{U2}(x) + O(1)$ because we can always emulate $U1$ on $U2$ and vice versa. This also holds for our two part code theory: $U1(\overline{U2}\ \bar{j}z) = x$ and $U2(\overline{U1}\ \bar{i}y) = x$. Suppose that $U2$ is much more efficient for generating x then $U1$. Then the conditions for the invariance theorem are there and we'll have $\overline{U2}\ \bar{j}z = \bar{i}y$, i.e. $\overline{U2} = \bar{i}$ and $\bar{j}z = y$. Under two-part code optimization we choose $\overline{U2}$ as the model for x. If we now change our choice of reference machine to $U2$. We choose by $U2(\bar{j}z) = x$ the program $\bar{j}$ as our model. Since $\overline{U2}\ \bar{j}z$ is optimal and thus incompressible $\overline{U2}$ and $\bar{j}$ have no mutual information. □

Lemma 2. *In the class of Turing equivalent system the size of the optimal models selected by means of two-part code optimization is not invariant.*

Proof: immediate consequence of theorem 1. Selecting another reference Turing machine possibly implies the emergence of a new optimal model that has no mutual information with the initial optimal model, this includes information about its size. □

The fact that universal machines pop up as optimal models prima facie seems undesirable. One might be tempted to argue that polysemy is caused by the extreme power of Turing equivalent systems and that one could control the effects by selecting weaker models of computation. It seems however that polysemy is a fairly natural condition. At any rate a restriction to total functions is not sufficient to rule out polysemy and vacuous models: there are hyperefficient total data splitting functions that give for every string a more efficient two-part representation that gives us 'meaningless' optimal models for free (See Paragraph 7). Consequently we make the following observation:

Observation 5 *Polysemy under two-part code optimization is a natural phenomenon in computational systems.*

Observation 5 explains how the results of [20] can be reconciled with those in [27]. There may be many models that exhaust the stochastic characteristics of a data set, even to the extent that the structural part of one model is the ad hoc part of another and vice versa. The choice for one model or another as the true model has consequences for the way the theory generalizes.

- Many computational functions that can be selected as optimal model for a data set under two-part code optimization are vacuous: i.e. they fit the definition of an optimal model but have no meaningful generalizations.

- In various cases universal Turing machines will be selected as optimal models. This is a direct consequence of the inveriance theorem of Kolmogorov complexity.
- Two-part code optimization in general computational systems is polysemantic: it may choose various different models, that have no mutual information, as optimal.

5 Conclusion

We give a short overview of the general results:

- Let p be a program, and x and y data sets, then the meta-sentence $'$”$p(x) = y$” *is true*$'$, specifies that the sentence ”$p(x) = y$” contains semantic information in the sense of Floridi's veridical theory: it contains well-formed data and it is truthful.
- The computation $p(x)$ specifies a *meaning* for y, provided that it really produces y as an output. Of course, in principle, there is an infinity of other possible computations that produce y and thus define a meaning for y.
- Moreover the program p can be interpreted as *instructional information*: it is true that when we follow the instructions specifed in p on the input x we get the output y. In this sense a computational theory of semantics is compatible with an *operational theory of meaning*: it specifies in an abstract sens the procedures we have to follow to make certain statements true.
- We get the *Kolmogorov complexity* $K(y)$ of y if we generalize $p(x) = y$ to $U(\overline{p}x) = p(x) = y$. Here U is a reference universal Turing machine and $\overline{p}$ a prefix-free code for the program p. We specify $K(y)$ as the length $l(p)$ of the smallest program p that generates y on an empty input $U(\overline{p}\epsilon) = p(\epsilon) = y$. By specifying an empty input ϵ we force *one-part code optimization.*
- We get *two-part code optimization* if we relax the condition that the program has to run on an empty input string ϵ. We can then study the shortest program $\overline{p}x$ that produces $U(\overline{p}x) = p(x) = y$. The condition of polysemy entails that there are other programs $\overline{q}z$, of comparable length, that produce y such that p and q have no, or little, mutual information. If the expression $\overline{p}x$ gives an optimal description of y, then $\overline{p}$, x, $\overline{q}$ and z are all random. In this case the programs p and q indeed capture all the relevant stochastic aspects of y, but the structural interpretation of y they imply is different.
- Consequently it is impossible to formulate an invariance proof for two-part code optimization. The models found may vary wildly with our choice of universal machine. In many cases even vacuous models, like small universal Turing machines and other general data processing routines, will be selected as 'optimal model'.
- This makes data compression a necessary, but not a sufficient, condition for optimal model selection. Data compression will work with high probability in situations where the data are collected in accordance with standard empirical methodology: several observations of the same phenomena under

various conditions.Ultimately the justification for model selection by data compression is empirical.

In this paper we have presented a theory of computation of semantics for data sets. We have shown how this theory is compatible with standard philosophical theories of semantic information. This does not imply, in itself, that our proposal is correct, but it suggests how a purely quantitative view on information can be reconciled with more qualitative interpretations. Computation deals with operations on data and in itself does not generate true facts, but we can specify the fact that certain operations on certain data lead to certain results. This is sufficient when we want to learn computational models in an empirical setting.

We can now answer the research question 1 from the beginning of this paper: we can use information theory to construct models more or less automatically on the basis of data sets, provided that we ensure that the data sets are gathered in accordance with the laws of empirical research. Applying data compression in the blind to ad hoc data sets is dangerous and will almost certainly lead to the selection of bad models.

6 Acknowledgements

I thank Steven de Rooij, Peter Bloem, Frank van Harmelen, Amos Golan, Erik Schultes, Peter van Emde Boas and my fellows at the Info-metrics Institute for many inspiring discussions. I thank the University of Amsterdam and especially Cees de Laat for allowing me the leeway to pursue my research interests. This research was partly supported by the Info-Metrics Institute of the American University in Washington, the Commit project of the Dutch science foundation NWO, the Netherlands eScience center, the IvI of the University of Amsterdam and a Templeton Foundations Science and Significance of Complexity Grant supporting The Atlas of Complexity Project.

7 Appendix: The Cantor function as a hyperefficient data splitting function

In this paragraph we prove that, given the equivalence between the set of binary strings and the set of natural numbers, there are total functions that for any constant c split sets of data that are dense in the set of all possible strings in to two parts, where the sum of the lengths of the subsets is at least c shorter than the length of the original data set. This goes against the 'folklore' conception of Kolmogorov complexity that the density of compressible strings is zero in the limit. It is, however, not in contradiction with fundamental results. The result is problematic for a platonic (ITP) interpretation of two-part code optimization: there are dense sets of data for which a 'meaningless' two-part code description exists. It is more efficient than one-part code and thus always should be considered as optimal model. In particular this result shows that a restriction to total functions is insuffienct to make two-part code optimization work properly.

The following basic observation holds: *In general two data sets contain more information than one.* This is immediately clear from the fact that when we cut a string of k bits into two pieces, we actually add $\log k$ bits of information about the location of the cut. Conversely we can code the information about a string of length c in two strings of length a and b such that $a + b < c$. This suggests that there might be mathematical functions that compress a data set by simply splitting it in two. Indeed there are many such functions in the domain of arithmetic. A special example is the socalled Cantor pairing function. The set of natural numbers $\mathbb{N}$ can be mapped to its product set by the Cantor pairing function $\pi^{(2)} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ that defines a two-way polynomial time computable bijection:

$$\pi^{(2)}(x, y) := \frac{1}{2}(x + y)(x + y + 1) + y \tag{7}$$

The Fueter - Pólya theorem [2] states that Cantor pairing is the only possible quadratic pairing function and the Fueter - Pólya conjecture states that there are no such higher order mappings. Note that there is an unlimited number of comparable bijections (e.g. Szudzik pairing)[1], but in terms of information conservation all these mappings stay asymptotically close to the Cantor function.

Lemma 3. *The Cantor pairing function $\pi^{(2)}(x, y)$ is information expanding on all values $(x, y), x, y > 0$. For any constant c the set of values (x, y) for which $pi^{(2)}(x, y)$ expands more than c bits is dense.*

Proof: Supose $x, y > 0$. We can measure the information efficiency as the logarithm of the balance of information in the input and the output of the function: $\log \pi^{(2)}(x, y) - \log x - \log y$ which gives:

$$\log \frac{\pi^{(2)}(x, y)}{xy} = \log \frac{\frac{1}{2}(x + y)(x + y + 1) + y}{xy} =$$

$$\log(\frac{x}{2y} + \frac{y}{2x} + \frac{3}{2x} + \frac{1}{2y} + 1) > \log 1 = 0$$

Take the set $\frac{x}{2y} > 2^c$ or $\frac{y}{2x} > 2^c$. For the domain as well as the range of the Cantor function these sets have a density of 2^{-c} in the limit. For elements of these dense sets $\log \pi^{(2)}(x, y) - \log x - \log y \leq c$. □

The ratio behind this result is the fact that the Cantor enumeration follows the counter diagonal $x + y = k$ and thus is highly inefficient close to the regions $x = 0$ and $y = 0$. This is the price one has to pay for a bijection between the two sets. The fact that the Cantor function and its symmetric counter part are also bijections implies that the inverse Cantor function is information compressing: we can split any positive natural number in two other positive numbers and compress information. This result shows that the intuition of Kolmogorov complexity that a string is random when it is incompressible and that the density of compressible strings is zero in the limit is wrong. The matter is more subtle:

[1] See http://szudzik.com/ElegantPairing.pdf, retrieved January 2016.

for every constant c there is a dense set of strings that can be compressed by at least c bits. Since it is dense such a set has typical elements that share a common quality and yet the elements of such a set are random. A 'valid' definition of randomness would thus need to invoke a much weaker notion of incompressibility: A set x is incompressible if $K(x) > l(x) - \log l(x)$. A set is random if it is incompressible. Kolmogorov complexity only gives us a rather sloppy approximation of the notion of randomness.

The consequences for two-part code optimization are important:

Lemma 4. *For every constant c there is in the set $\{1, 0\}^*$ of all finite strings a dense subset for which two-part code compression is c bits shorter than one-part code compression. The associated algorithmic statistics are trivial and do not describe any stochastic qualities of the data set.*

Proof: take $n = \pi^{(2)}(a, b)$. By lemma 3 there is for every constant c a dense set of pairs (a, b) for which $\log a + \log b = \log n - c$. Define $S_a = \{\pi^{(2)}(a, y) | \ 0 < y \leq b\}$ and $S_b = \{\pi^{(2)}(x, b) | \ 0 < y \leq a\}$. Note that $|S_b| = a$ and $|S_a| = b$. We show that these sets meet the model criteria for algorithmic sufficient statistics in definition 9. We give the proofs for S_a: 1) Sufficient Statistics: We have $K(n|S_a) = \log |S| + O(1)$. If we know S_a then we only need an index of length $\log |S_a| = \log b$ to identify b, which allows us to compute $n = \pi^{(2)}(a, b)$ with the code for $\pi^{(2)}$ of length $O(1)$. 2) Symmetry: $K(n, S_a) = K(n) + O(1)$. If we have n and the code for $\pi^{(2)}$ of length $O(1)$ then we can compute (a, b) and thus S_a. In this construction any number can serve as a model and any number can be modeled, so the models have no explanatory power. □.

As a consequence:

Lemma 5. *Two-part code optimization is polysemantic for total functions.*

Proof: Immediate consequence of the proof of lemma 4. The Cantor functions are bijections on $\mathbb{N}$ so total by definition. Both S_a and S_b are models but since the sets are dense they will contain dense subsets of incompressible numbers. For these numbers a and b will have no mutual information. □

Bibliography

[1] Gottlob Frege. Begriffsschrift: eine der arithmetischen nachgebildete Formelsprache des reinen Denkens. Halle, 1879.

[2] Rudolf Fueter, Georg Pólya: Rationale Abzhlung der Gitterpunkte, Vierteljschr. Naturforsch. Ges. Zrich 58 (1923), Pages 280386.

[3] G.D. Birkhoff: Collected Mathematical Papers, New York: American Mathematical Society, 1950.

[4] Rényi, Alfréd (1961). "On measures of information and entropy". Proceedings of the fourth Berkeley Symposium on Mathematics, Statistics and Probability 1960. pp. 547561.

[5] Bar-Hillel, Y., 1964, Language and Information: Selected Essays on Their Theory and Application, Reading, Mass; London: Addison-Wesley.

[6] Ray. J. Solomonoff. A formal theory of inductive inference: Parts 1 and 2. Information and Control, 7:1–22 and 224–254, 1964.

[7] Max Bense: Aesthetica. Einführung in die neue Aesthetik. Baden-Baden: Agis-Verlag, 1965.

[8] de Groot, Adrianus Dingeman. Methodology: Foundations of inference and research in the behavioral sciences. MTH, 1969.

[9] J. J. Rissanen, (1978) Modeling by Shortest Data Description, Automatica, volume 14, no. 5, pg. 465-471.

[10] M. Koppel, (1987) Complexity, Depth, and Sophistication", in Complex Systems 1, pages = 1087-1091.

[11] C. H. Bennett, (1988) Logical depth and physical complexity. In R. Herken, editor, The Universal Turing Machine: A Half-Century Survey, pages 227-257. Oxford University Press.

[12] J. J. Rissanen, (1989) Stochastic Complexity in Statistical Inquiry, World Scientific, Singapore.

[13] J.P. Crutchfield and K. Young, (1989) Inferring Statistical Complexity. Physical Review Letters 63:105.

[14] J.P. Crutchfield and K. Young, (1990) Computation at the Onset of Chaos, in Entropy, Complexity, and the Physics of Information, W. Zurek, editor, SFI Studies in the Sciences of Complexity, VIII, Addison-Wesley, Reading, Massachusetts. pp. 223-269.

[15] R. Scha and R. Bod (1993) "Computationele Esthetica", Informatie en Informatiebeleid 11, 1 (1993), pp. 54-63.

[16] J.P. Crutchfield (1994) The Calculi of Emergence: Computation, Dynamics, and Induction, Physica D 75, pg. 11-54.

[17] M. Koppel (1995) Structure, The universal Turing machine (2nd ed.): a half-century survey. pages 403-419. Springer Verlag.

[18] Ray R. Solomonoff, "The Discovery of Algorithmic Probability", Journal of Computer and System Sciences, Vol. 55, No. 1, pp. 73-88, August 1997.

[19] Pedro Domingos. Occam's Two Razors: The Sharp and the Blunt. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (pp. 37-43), 1998. New York, NY: AAAI Press.

[20] Vereshchagin N.K., Vitányi P.M.B., Kolmogorov's structure functions and model selection, IEEE Trans. Information Theory, vol. 50, nr. 12, 3265–3290, (2004)

[21] N. Chater and P. Vitányi , Simplicity: A unifying principle in cognitive science? Trends in Cognitive Sciences, 7:1(2003), 19–22.

[22] R. Cilibrasi, P.M.B. Vitanyi, Clustering by compression, IEEE Trans. Information Theory, 51:4(2005), 1523- 1545. http://xxx.lanl.gov/abs/cs.CV/0312044 (2003)

[23] M. Gell-Mann and S. Lloyd (2003) Effective complexity. In Murray Gell-Mann and Constantino Tsallis, eds. *Nonextensive entropy–Interdisciplinary applications*, Oxford University Press, 387-398.

[24] J. W. McAllister, (2003) Effective Complexity as a Measure of Information Content, Philosophy of Science, Vol. 70, No. 2, pp. 302-307.

[25] L. Antunes and L. Fortnow, (2003) Sophistication Revisited. In Proceedings of the 30th International Colloquium on Automata, Languages and Programming, volume 2719 of Lecture Notes in Computer Science, pages 267-277. Springer.

[27] Wolpert, D.H., and Macready, W.G. (2005) "Coevolutionary free lunches," IEEE Transactions on Evolutionary Computation, 9(6): 721735

[28] P.M.B. Vitányi, (2006) Meaningful information, IEEE Trans. Inform. 52:10, 4617 - 4626.

[29] L. Antunes, L. Fortnow. D. Van Melkebeek and N. V. Vinodch, (2006) Computational depth: Concept and application, Theoretical Computer Science, volume, 354.

[30] Cover T.M. and Thomas, J.A. (2006) Elements of Information theory, Wiley.

[31] D.H. Wolpert and W. Macready (2007) Using self-dissimilarity to quantify complexity: Research Articles, Complexity, volume 12,number 3, pages 77–85.

[32] Grünwald, P. (June 2007). "the Minimum Description Length principle". MIT Press.

[33] Li M., Vitányi P.M.B. (2008), An Introduction to Kolmogorov Complexity and Its Applications, 3rd ed., Springer-Verlag, New York.

[34] P.W. Adriaans, The philosophy of learning, the cooperative computational universe, in Handbook of Philisophy of Information, P.W.Adriaans, J.F.A.K. van Benthem (eds.), Elseviers Science Publishers, 2008

[35] P.W. Adriaans, Between Order and Chaos: The Quest for Meaningful Information, Theory of Computing Systems, Volume 45 , Issue 4 (July 2009), Special Issue: Computation and Logic in the Real World; Guest Editors: S. Barry Cooper, Elvira Mayordomo and Andrea Sorbi Pages 650-674, 2009.

[36] P. W. Adriaans and P. M. B. Vitányi, (2009) Approximation of the Two-Part MDL Code, Comput. Sci. Dept., Univ. of Amsterdam, Amsterdam; Information Theory, IEEE Transactions on, Volume: 55, Issue: 1, On page(s): 444-457.

[37] P.W. Adriaans , A critical analysis of Floridi's theory of semantic information, In Knowlegde, Technology and Policy, Hilmi Demir ed. : Luciano Floridi's Philosophy of Technology: Critical Reflections, 2010.

[38] Samuel Rathmanner and Marcus Hütter, A Philosophical Treatise of Universal Induction, Entropy 2011, 13(6), 1076-1136.

[39] L. Floridi, 2011, The Philosophy of Information, Oxford; Oxford University Press.

[40] P..W. Adriaans, Facticity as the amount of self-descriptive information in a data set, arXiv:1203.2245 [cs.IT], 2012.

[41] Luís Antunes, Andre Souto, and Andreia Teixeira. 2012. Robustness of logical depth. In Proceedings of the 8th Turing Centenary conference on Computability in Europe: how the world computes (CiE'12), S. Barry Cooper, Anuj Dawar, and Benedikt Lwe (Eds.). Springer-Verlag, Berlin, Heidelberg, 29-34.

[42] Peter Bloem, Francisco Mota, Steven de Rooij, Luis Antunes, Pieter Adriaans: A Safe Approximation for Kolmogorov Complexity. ALT 2014: 336-350.

[43] Bloem P., de Rooij S., Adriaans P. (2015) Two Problems for Sophistication. In: Chaudhuri K., GENTILE C., Zilles S. (eds) Algorithmic Learning Theory. Lecture Notes in Computer Science, vol 9355. Springer.

[44] Spade, Paul Vincent and Panaccio, Claude, ”William of Ockham”, The Stanford Encyclopedia of Philosophy (Winter 2016 Edition), Edward N. Zalta (ed.), URL = ¡https://plato.stanford.edu/archives/win2016/entries/ockham/¿.

[45] Floridi, Luciano, ”Semantic Conceptions of Information”, The Stanford Encyclopedia of Philosophy (Spring 2017 Edition), Edward N. Zalta (ed.), URL = ¡https://plato.stanford.edu/archives/spr2017/entries/information-semantic/¿.