2012 International Conference on Computer Networks and Communication Systems (CNCS 2012)
IPCSIT vol.35(2012) © (2012) IACSIT Press, Singapore

On the Algorithmic Complexity of Vague Descriptions

Christian Fermüller

Vienna University of Technology, Austria

Abstract. We propose a fuzzy logic based model of vague descriptions that refers to a variant of Kolmogorov complexity. The model supports the quantification of the decrease of information associated with the decrease of precision entailed by the vagueness of (electronically communicated) descriptions. Our results address the challenge to model efficient communication with vague predicates and to connect it to the literature on 'theories of vagueness', but also to classical concepts of information theory.

Keywords: fuzzy logic, vagueness, Kolmogorov complexity, quantifying information

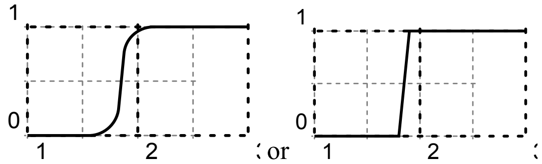
1. Introduction

Vagueness is a ubiquitous phenomenon. Models and measures of information that can cope (also) with the transmission of vague information are still rare. On the other hand, it is frequently and convincingly argued that the transmission of vague descriptions is *adequate* and *expedient* in many scenarios, due to the decreased amount of spurious information compared to unnecessarily precise descriptions. Many formal approaches to reasoning with vague notions and predicates are debated in the literature (for an overview see, e.g., [1,7,8,9,14]). However, hardly any of the proposed theories of vagueness seems to assist the *quantification* of the decrease of information induced by a decrease of the level of preciseness imposed on descriptions of intrinsically complex objects. We present a simple mathematical model of vague descriptions as fuzzy descriptions, based on a variant of Kolmogorov complexity, that addresses this problem and that can be linked to various wider frameworks for handling vague information.

2. The Enigma of Quantifying Vague Information

The popularity of *fuzzy logic* demonstrates that in many contexts a direct, formal, and systematic treatment of graded predicates and degrees of truth, intended to model vagueness, is possible and useful. An often cited reason for tackling vague information head on, instead of trying to eliminate it from the object language in question, is that it is deemed *less costly* to convey imprecise and partial information than to replace it by unnecessarily precise information before communication. Indeed, *spurious precision* may not only be wasteful in terms of increased complexity, but is also potentially harmful, since it may invite inadequate interpretations of data that should be processed according to their intended level of precision. For example, in reply to an inquiry about the general direction to follow in order to reach the inner city of a large town, we neither expect nor want to be told a precise numerical value specifying a vector pointing to the exact city center from some precisely fixed point of reference. (Neither do we want to learn the precise *interval* of all degrees of vectors that intersect the boundaries of the inner district of the town.) A *vague* gesture from a well informed and truthful responder might well be the most adequate and useful information in this context. Similarly, when we ask about John's current height, 'rather short' might be a more adequate reply than '1651 millimeters measured using device X on 08:13am on April 1st, 2008', even if this is indeed the most reliable and precise information about John's height.

Fuzzy logic suggests to specify the semantics of predicates like 'short', 'tall', 'old', etc. by reference to fuzzy sets, i.e., to maps from the universe of discourse into the real closed unit interval [0,1], instead of ordinary ('crisp') sets. Indeed is not hard to understand that typically used maps like



assigning truth values from $[0,1]$ to possible heights (here in meter), result in more refined models of the predicate 'tall' than any assignment of classical truth values 0 and 1 to the same range of possible heights. However, such models seem to sever the link to the above mentioned motivation for using vague language, namely, that vague information is usually less costly to generate and to transmit than corresponding precise information. Obviously, this is at variance with the trivial observation that *more*, not less, bits are needed in general to communicate truth values or membership degrees drawn from $[0,1]$ rather than from $[0,1]$.

Fuzzy logicians might reply that their formal tool box of fuzzy sets and relations, t-norm based logics, graded consequence relations, etc., is not intended to model the indicated reduction of information enabled by vague language. Rather fuzzy logic is seen as a formal model of reasoning with vague information that successfully explains *other phenomena* and puzzles attributed to vagueness, in particular the *sorites paradox*, see, e.g., [6]. Nevertheless one may ask for a related formalism that straightforwardly models the perceived decrease of information connected with the transition from precise to fuzzy descriptions of intrinsically complex structures and objects. Moreover, fuzzy logic is by no means the only formal model of reasoning with vague notions and propositions. In fact, there are very prolific debates in contemporary analytic philosophy on 'theories of vagueness' that frequently oppose the degree based approach exemplified by fuzzy logic. We refer to [1,8] for more information on this debate. Here we have to confine ourselves to a few remarks in Sections 4 that attempt to show that our proposed model matches insights arising from various competing approaches to vagueness. In any case, it is significant that also supervaluation, epistemicism, contextualism, as well as pragmatic theories of vagueness do not offer tools that allow to quantify the loss of information involved in the transition from more precise to less precise specifications.

In conclusion to these observations, one may speak of an *enigma of quantifying vague information*, consisting in the antagonism between two facts: (1) The preference for fuzzy descriptions over unnecessarily precise ones can be motivated informally by a decrease in complexity induced by an intentional loss of irrelevant information. (2) Neither degree based nor alternative formal models of reasoning with vague information seem to provide a clear basis for quantifying the corresponding decrease of complexity and information. We augment this motivation by listing some *desiderata* for an adequate model of complexity of fuzzy descriptions. We aim at a simple, mathematically elegant model of fuzzy descriptions that captures the outlined intuition that reduced levels of precision correspond to decreased amounts of (algorithmic/ descriptive) complexity, is robust with respect to the choice of underlying description languages, measures information in bits, as usual, and is compatible with existing models of descriptive complexity and information: 'crisp' descriptions should appear as an obvious limit case of (variously precise) fuzzy descriptions.

3. Algorithmic Complexity of Crisp Descriptions

There already exists an elegant theory for quantifying the complexity of *precise* descriptions, i.e., of complete and unambiguous specifications of formal objects: it is variously known as 'algorithmic information theory', 'Kolmogorov(-Chaitin) complexity', 'program-size complexity', 'minimum description length' and some other names. Here we follow the terminology of the well known monograph [10]. The *Kolmogorov complexity* $C(x)$ of a binary string x intends to capture the complexity inherent in *any* precise description of x. Informally, $C(x)$ is maximal if there is no way to describe x without explicitly listing all bits of which x consists. On the other hand, $C(x)$ is small, even for a very long string x, if x can be precisely specified by a short description. To put this idea to work we fix a universal Turing machine U and define $C_U(x)$ as the length of the shortest program p (Turing machine) for U that outputs x. As usual, we identify $\{0,1\}^*$ with the set of natural numbers □ using the lexicographical ordering on binary strings. Hence U computes a universal partial recursive function.

Definition 1 *The (conditional) Kolmogorov complexity* $C_U(x|y)$ *of x given y, in reference to a universal Turing machine U, is defined by* $C_U(x|y)=\min\{l(p):U(\langle p,y\rangle)=x\}$ *where* $l(p)$ *is the length of p and* $\langle\cdot,\cdot\rangle$ *is the standard recursive bijective pairing function. We write* $C_U(x)$ *for* $C_U(x|\varepsilon)$.

We drop the index denoting the *reference machine U* and simply write $C(x|y)$, whenever we are only interested in $C_U(x|y)$ up to an additive constant. This is justified by the well known fact that any two universal Turing machines can simulate each other. More exactly, for all x and y we have $|C_U(x|y)-C_V(x|y)|\le c$ for a constant c that depends only on the reference machines U and V.

Note that in identifying *descriptions D* with programs we implicitly assume that the described objects are effectively specified as certain binary strings. Again, this is justified by the outlined robustness of the concept. E.g., in claiming that the Kolmogorov complexity of a random (incompressible) directed graph G_n with n vertices is $2\log(n)+O(1)$ we may refer to any standard format in which the graph is to be presented (e.g., as binary $n\times n$-matrix, a list of edges consisting of pointers to vertices, or as a lexicographically sorted list of pairs of connected vertices, etc.). We may safely assume that for any two standard formats ξ and ξ' of presenting graphs there is a program that converts, for any given graph G, the binary presentation $\xi(G)$ into the binary presentation $\xi'(G)$. It follows that $C(\xi'(G))=C(\xi(G))+O(1)$ and consequently we may, by slight abuse of notation, simply write $C(G_n)=2\log(n)+O(1)$.

4. Probabilistic Programs as Dispersive Descriptions

In speaking about a fuzzy description $\tilde{D}$ we will assume that there is some (in principle) precisely specifiable object x that is imprecisely described by $\tilde{D}$. Again, we may assume without essential loss of generality that x is coded by a binary string. However in contrast to Section 3, above, we cannot assume that x can be uniquely reconstructed from $\tilde{D}$ without further information. In other words, we take vague descriptions to convey only *partial information* about the described object.

Another feature of vague information, besides its partiality, consists in its *dispersiveness*. In this analysis we follow Robin Giles [2,3,4], who already in the 1970s presented a model for reasoning with fuzzy propositions and notions that combines a Lorenzen style dialogue game with a scheme for evaluating atomic sentences that refers to results of dispersive binary (yes/no) experiments. I.e., these semantic tests may show different results upon repetition; only a certain *probability* of yielding a positive answer is known. In our context, where descriptions of an object x are identified with programs that compute (a binary presentation of) x, this motivates the generalization from deterministic Turing machines, that compute a single binary string as output, to *probabilistic Turing machines* (PTMs) that produce dispersive output. The same PTM $\tilde{T}$ may, for the same input y, produce different output strings, every time $\tilde{T}$ is invoked. We only care about the case where some state transitions may randomly lead to one of two alternative states with equal probability. Such PTMs may be represented as deterministic Turing machines that have access to an additional *random tape* filled with random, uniformly distributed 0s and 1s. (Alternatively, one may add an additional instruction for random branching. Details of the representation will not matter here.) For any PTM $\tilde{T}$, $\Pr[T(x)=y]$ denotes the probability that $\tilde{T}$ outputs y on input x. Note that a terminating deterministic Turing machine T can be viewed as a PTM where for any input x there exists a y such that $\Pr[T(x)=y]=1$ (and consequently $\Pr[T(x)=z]=0$ for all $z\neq 0$).

In analogy to the classical case of Section 3, the intended version of algorithmic complexity theory relies on the following *invariance theorem:*

Theorem 2 *Let* $\tilde{T}_1,\tilde{T}_2,\ldots$ *be an effective enumeration of all PTMs, understood as programs with binary input and (probabilistic) output. Then there exists a* universal PTM $\tilde{T}_U=\tilde{U}$ *among them, such that for all* $i>1$ *and all x and y we have* $\Pr[\tilde{U}(\langle i,y\rangle)=x]=\Pr[\tilde{T}_i(y)=x]$.

Of course it were inappropriate to define $C_{\tilde{U}}(x|y)$ simply as the length of the smallest PTM that computes x given y with some positive probability: it is easy to specify a fixed, small PTM that outputs *all x* on any input with some positive probability. Indeed, we still have to argue that the 'dispersive output' of a PTM $\tilde{T}$ can be usefully interpreted as the set of possible, but vague descriptions of a single object. To this aim, note that we can view $\tilde{T}$'s output as a *fuzzy singleton set*, i.e., a set with σ-count 1 (see, e.g., 11). More precisely, remember that a fuzzy set F is a function from some universe X into $[0,1]$; where in our case

$X = \{0,1\}^*$. The *cardinality* or σ-*count* of *F* is defined as $\#(F) = \sum_{x \in X} F(x)$, where $F(x)$ denotes the membership degree of *x* in *F*. By interpreting the probability $\Pr[\tilde{T}(y) = x]$ of obtaining *x* on input *y* as a membership degree for *x*, we identify the output $\tilde{T}(y)$ with a fuzzy set where $\#(\tilde{T}(y)) = 1$.

Viewing the output of a PTM $\tilde{T}$ as a fuzzy singleton set fits the idea of identifying $\tilde{T}$ with a fuzzy description, in the sense of a mechanism that provides 'dispersive presentations' of some finite object. However it does not determine *which* object is actually described in this way. Whereas in the special case of deterministic programs a unique object is presented for any given input, $\tilde{T}$ (given input *y*) can in general be read as vague description of *different* objects. If we want to interpret the output x' of $\tilde{T}$ as fuzzy description of a specific object *x* at some level of precision, we need to assess how much information about *x* is contained in x'. Fortunately, Kolmogorov complexity provides a good candidate for a corresponding measure: we will identify the information about *x* contained in x' with the length of the smallest program that converts *x* into x', i.e., we use $C_U(x|x')$ for this purpose. This leads to the following definition of the overall *closeness* of the (dispersive) output of a PTM to a given *x* as the expected number of bits needed to convert the output into *x*.

***Definition* 3** *Let $\tilde{U}$ be a universal PTM and let p be a program for $\tilde{U}$, i.e., a standard representation of an arbitrary PTM. Moreover let U be a deterministic universal TM. Then the* computational closeness to *x* of the output of *p* on input *y is given by* $ccl_{\tilde{U},U}(x|p,y) = \sum_{z \in \{0,1\}^*} \Pr[\tilde{U}(\langle p,y \rangle) = z] \cdot C_U(x|z)$.

Note that *two* universal reference machines are involved in the above definition: a probabilistic one to fix the semantics of *p*, and a deterministic one that fixes the number of bits needed to convert any given output of *p* on *y* to *x*. Since deterministic TMs are special cases of PTMs one may use the same universal PTM in both instances. More importantly, invariance - as implied for PTMs by Theorem 2 - entails that the reference to the underlying universal machines can be dropped, as long as we are only interested in this quantity up to an additive constant. We may assume reference machine *U* to respect the principle that for any given *x* no string *y* contains less information about *x* than the empty string, i.e., $C_U(x|z) \le C_U(x)$. Consequently we obtain that $\ell = 1 - ccl_{\tilde{U},U}(x|p,y) / C_U(x|y)$ is a value in $[0,1]$. We will call ℓ the *precision level* of the 'fuzzy description' *p* of *x* given *y*. If $\ell = 1$ then $ccl_{\tilde{U},U}(x|p,y) = 0$, implying that no additional information is needed to obtain *x* from *p* and *y*. In other words: *p* (given *y*) can be considered a *fully precise* description of *x*. On the other extreme, if $\ell = 0$ then $ccl_{\tilde{U},U}(x|p,y) = C_U(x|y)$, which implies hat *p* does not contain any information about *x* that is not already contained in *y*. In that case we call *p completely vague* with respect to *x*, by which we mean that it is as useless for reconstructing *x* as is the empty program.

The just introduced concept finally allows us to present a definition of the algorithmic information $\tilde{C}^{\pi}(x|y)$ contained in a binary string *x* at a certain precision level π, given *y*.

Definition 4 *For* $0 \le \pi \le 1$ *the* probabilistic Kolmogorov complexity at precision level π *of x given y is defined by* $\tilde{C}^{\pi}_{\tilde{U},U}(x|y) = \min\{l(p) : 1 - ccl_{\tilde{U},U}(x|p,y) / C_U(x|y) \ge \pi\}$. *We write* $\tilde{C}^{\pi}_{\tilde{U},U}(x)$ *for* $\tilde{C}^{\pi}_{\tilde{U},U}(x|\varepsilon)$.

5. Basic Properties of $\tilde{C}^{\pi}$

It remains to check whether the proposed model satisfies the *desiderata* listed at the end of Section 2. Note that it follows immediately from Definition 4 that the probabilistic Kolmogorov complexity of any object increases monotonically with the imposed precision level: $\pi \ge \pi'$ implies $\tilde{C}^{\pi}_{\tilde{U},U}(x|y) \ge \tilde{C}^{\pi'}_{\tilde{U},U}(x|y)$. In consequence, the intuition that higher levels of vagueness allow for less complex descriptions is indeed respected in our model.

The following theorem expresses the *robustness* of $\tilde{C}^{\pi}_{\tilde{U},U}(\cdot|\cdot)$:

Theorem 5 *For all universal PTMs $\tilde{U}$, $\tilde{U}'$, and deterministic TMs U, U' there exists a constant d such that for all precision levels* $\pi \in [0,1]$ *and all* $x, y \in \{0,1\}^*$: $\tilde{C}^{\pi}_{\tilde{U},U}(x|y) \le \tilde{C}^{\pi}_{\tilde{U}',U'}(x|y) + d$, *where d only depends on* $\tilde{U}$, $\tilde{U}'$, *U, U'.*

Again this justifies dropping the reference to underlying universal machines. The most fundamental property of our notion of probabilistic Kolmogorov complexity can succinctly be expressed as follows:

Theorem 6 *For all precision levels* π *and all* $x, y \in \{0,1\}^*$: $\tilde{C}^{\pi}(x|y) = \pi \cdot C(x|y) + O(1)$.

As immediate consequences we obtain some facts confirming that $\tilde{C}^{\pi}(\cdot)$, viewed as a measure of information contained in fuzzy descriptions, matches further basic intuitions:

- $\tilde{C}^{1}(x|y) = C(x|y) + O(1)$, which confirms that for precision level 1 probabilistic Kolmogorov complexity coincides with ordinary Kolmogorov complexity up to an additive constant. If the same universal reference machine is used for $\tilde{C}^{\pi}$ and C, respectively, and if $C(x|x) = 0$ (i.e., copying input into output is discounted) then one gets rid of the additive constant.
- $\tilde{C}^{0}(x|y) = O(1)$, which means that at precision level 0 all specific information is lost.
- $\tilde{C}^{0.5}(x) = \log_2(x)/2 + O(1)$ for incompressible x, implying that (up to an additive constant) half of the relevant information lost at precision level 0.5, or, to put it differently: half of its bits are needed to describe an incompressible string at precision level 0.5.

6. Conclusion

We have illustrated that a straightforward generalization of Kolmogorov complexity to probabilistic programs leads to a model of vague descriptions of (binary coded) objects supporting the quantification of information contained in such descriptions. This provides a solution to a conceptual problem regarding the transmission of vague information, referred to as the *enigma of quantifying vague information*: the perceived intended loss of information due to vagueness cannot easily be represented in standard models of vague expressions. However the offered solution is at best partial and preliminary, as long as it is not properly placed in the context of the ongoing discourse on appropriate `theories of vagueness'. Further research is needed to assess whether our concept indeed usefully augments standard models and techniques in approximative reasoning and information processing.

7. Acknowledgements

Partially supported by FWF grant~I143-G15 (Eurocores-LogICCC project LoMoReVI).

8. References

[1] P. Cintula, C.G. Fermüller, L. Godo, P. Hájek (eds.): Understanding Vagueness -- Logical, Philosophical and Linguistic Perspectives. College Publications, 2011.

[2] R. Giles: A non-classical logic for physics. Studia Logica 33, vol. 4, (1974), 399-417.

[3] R. Giles: Łukasiewicz logic and fuzzy set theory. Int. J. Man-machine Studies 8, 1976, 313-327.

[4] R. Giles: Semantics for fuzzy reasoning. Int. J. Man-machine Studies 17, 1982, 401-415.

[5] P. Hájek: Metamathematics of Fuzzy Logic. Kluwer, 1998.

[6] P. Hájek, V. Novák: The sorites paradox and fuzzy logic. Int. J. General Systems 32, 2003, 373-383.

[7] D. Hyde: Logics of Vagueness. In: D.M. Gabbay, J. Wood (eds.), Handbook of the History of Logic, Vol. 8, North-Holland/Elsevier, 2007, 285-324.

[8] R. Keefe: Theories of Vagueness, Cambridge University Press, 2000.

[9] R. Keefe, P. Smith (eds.): Vagueness: A Reader, Massachusetts, MIT Press, 1987.

[10] M. Li, P. Vitányi: An Introduction to Kolmogorov Complexity and its Applications. Springer, 2008.

[11] H.T. Nguyen, E.A. Walker: A First Course in Fuzzy Logic. 3rd edition, Chapman & Hall/CRC, 2006.

[12] V. Novak, I. Perfilijeva, J. Mockor: Mathematical principles of fuzzy logic, Kluwer, 1999.

[13] M. Pinkal: Logic and Lexicon The Semantics of the Indefinite. Springer, 1995.

[14] T. Williamson: Vagueness, London, Routledge, 1994.

[15] L.A. Zadeh: Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans. Syst. Man and Cybern. 1, 1975, 28-44.