Data-dependence of Semantic Information

J Heidema[1] and W A Labuschagne[2]

[1] *Department of Mathematics, Rand Afrikaans University, P.O. Box 524, Johannesburg, 2000.*

[2] *Department of Computer Science, University of South Africa, P.O. Box 392, Pretoria, 0001.*

Abstract

The theory of semantic information is sketched in an algebraic setting, and automorphisms characterising various degrees of incompatibility are introduced. Data-dependent measures of semantic information are defined in terms of these automorphisms.

Keywords: *information, probability, logic, clause, Boolean algebra, automorphism*

Computing Review Categories: *F.4, G.2, G.3, H.1*

Received August 1989, Accepted April 1990.

1. The Theory of Semantic Information

As suggested by the word "semantic", this theory embodies an attempt to determine the (amount of) information contained in a statement by taking the meaning of the statement into account. At its present stage of development, the theory is able to deal with statements of formal propositional logic only. The resemblance between a formal propositional language and a programming language like PROLOG renders the restriction perhaps less irksome to computer scientists than to philosophers.

A propositional language consists of strings built up from *sentence symbols* like "p" and "q" with the aid of formula-building functions which prefix or infix certain additional symbols. For our purposes it is convenient to consider the following three formula-building functions:

- *conjunction*, which infixes the symbol "∧" between two strings;
- *disjunction*, which infixes "∨" between two strings;
- *negation*, which prefixes "¬" to a string.

Of course, these symbols represent the natural language connectives "and", "or" and "not" respectively.

A *valuation* of the language (also called an *interpretation*) is an assignment of truth values from the set {T,F} to sentence symbols, one value for each symbol. Intuitively, a valuation v with, say, v(p) = F and v(q) = T, has the effect of choosing, from amongst all the possible worlds we might be talking about, the one world in which the atomic facts are that p is false and q is true. (For simplicity, we assume that "p" and "q" are the only sentence symbols in our language, although everything we say can be generalised to the case of n sentence symbols.) In a well-known way, a valuation can be extended so as to assign a truth value to every string of the language. Many strings prove to be *semantically equivalent* in the sense that no valuation can distinguish between them. For example, the strings p∨q and q∨p always get truth values that coincide, no matter what values the symbols p and q are assigned. We may regard two semantically equivalent strings as different forms of the same *proposition*. If we start with the sentence symbols p and q, there are only sixteen propositions in the language (see table 1) whereas there are of course infinitely many strings in the language. (In the table we have used three connectives that are definable in terms of ∧, ∨ and ¬, namely → (if...,then...), ↔ (if and only if), and + (exclusive or).)

Familiar form	Complete clausal form
p∨¬p	no clauses
p∨q	{p,q}
q→p	{p,¬q}
p→q	{¬p,q}
¬p∨¬q	{¬p,¬q}
p	{p,q}, {p,¬q}
q	{p,q}, {¬p,q}
p+q	{p,q}, {¬p,¬q}
p↔q	{p,¬q}, {¬p,q}
¬q	{p,¬q}, {¬p,¬q}
¬p	{¬p,q}, {¬p,¬q}
p∧q	{p,q}, {p,¬q}, {¬p,q}
p∧¬q	{p,q},{p,¬q}, {¬p,¬q}
¬p∧q	{p,q}, {¬p,q}, {¬p,¬q}
¬p∧¬q	{p,¬q}, {¬p,q}, {¬p,¬q}
p∧¬p	{p,q}, {p,¬q}, {¬p,q}, {¬p,¬q}

Table 1

The popularity of resolution and PROLOG means that most readers will be familiar with the representation of propositions as sets of clauses. If the reader is unfamiliar with clausal form logic, he need merely recall that every proposition can be written in conjunctive normal form, that is, as a conjunction of disjunctions of literals, where by a literal we understand a sentence symbol or the negation of a sentence symbol. A proposition in conjunctive normal form can then be altered to clausal form by replacing the conjunction symbols by commas and rewriting the disjunctions of literals as sets of literals. For example, the proposition p has the conjunctive normal form $(p \vee q) \wedge (p \vee \neg q)$, and hence (omitting outer braces) the clausal form $\{p,q\}$, $\{p,\neg q\}$ (as indicated in table 1).

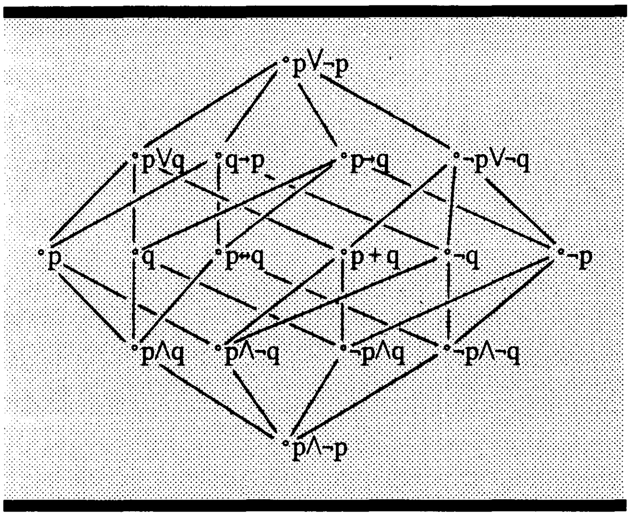

Figure 1

A slight refinement is that we have expressed our propositions in *complete clausal form*, that is as a set of clauses each of which contains, for every sentence symbol, either that symbol or its negation. This is no coincidence. The logically weakest non-tautological propositions are the complete clauses. To illustrate, we give in figure 1 the Hasse diagram for our propositional language with p and q as sentence symbols; if a proposition x lies below y and is connected to y by a sequence of one or more edges, then x *semantically entails* y, that is, every valuation assigning the value T to x will assign the value T to y also. The propositions that lie on the level just below the top are the complete clauses (as may be verified from table 1). Clearly a complete clause is logically weaker than any proposition lying lower down, in the sense that the complete clause semantically entails fewer propositions. We follow Carnap and Bar-Hillel [3] in taking the complete clauses to be our *content-elements*, that is propositions that carry unit information, and we define the *semantic information carried by a proposition* to be the *complete clausal form of that proposition*, that is, the set of content-elements semantically entailed by the proposition. For a given proposition x, call this set CONT(x).

The *amount of information* conveyed by a proposition x is defined to be $\mathrm{cont}(x) = k/m$, where k is the number of clauses in the complete clausal form of x and m is the total number of complete clauses in the language. (If the language is generated by sentence symbols $p_1, p_2, \ldots, p_n$ then $m = 2^n$.) The values of cont(x) for our illustrative language are given in table 2.

There is a simple relationship between cont and probability. The greater the *logical probability* of x, the less the amount of information conveyed by proposition x.

Theorem 1:
If prob(x) is defined to be r/m, where r is the number of valuations v such that $v(x) = T$ and m is the total number of valuations of the language, then prob(x) is a probability measure and $\mathrm{cont}(x) = 1 - \mathrm{prob}(x)$.

x	cont(x)
$p \vee \neg p$	0
$p \vee q$	1/4
$q \rightarrow p$	1/4
$p \rightarrow q$	1/4
$\neg p \vee \neg q$	1/4
p	1/2
q	1/2
p+q	1/2
$p \leftrightarrow q$	1/2
$\neg q$	1/2
$\neg p$	1/2
$p \wedge q$	3/4
$p \wedge \neg q$	3/4
$\neg p \wedge q$	3/4
$\neg p \wedge \neg q$	3/4
$p \wedge \neg p$	1

Table 2

Remarks:

- Our choice of the same symbol, m, for the denominator in the definitions of both cont and prob is deliberate: the total number of valuations of a language generated by n sentence symbols is also 2^n. Theorem 1 now follows from the fact that $k = m-r$.
- Carnap and Bar-Hillel define, in [3], for each logical probability measure (more accurately, for each "proper m-function m_p" where m is now something else) a corresponding content-measure function by setting $\mathrm{cont}(x) = m_p(\neg x)$. It is easy to show that $m_p(\neg x) = 1 - m_p(x)$. The function prob defined above is precisely the probability measure called c† by Carnap in [2]. We have preferred in the present exposition to limit ourselves to a single (and, in our opinion, the most natural) probability measure.

2. Data-dependence of Information

It is clear that the information effectively conveyed by a statement depends on a variety of factors. Rzevski [6] points out one of these: "For example, if someone addresses you in Japanese, the amount of Information you will be able to extract from this communication will depend on your knowledge of Japanese language, customs, conventions, gestures and facial expressions as well as on the situation in which the conversation takes place...". These subjective or psychological forms of context-dependence lie outside the scope of the theory of semantic information. For the purposes of this theory we postulate an ideal receiver, who possesses all relevant knowledge about the language, and regard the information conveyed by a statement as something inherent in the statement. There are nevertheless objective forms of context-dependence which may be investigated.

Suppose we have certain information stored in a database. A statement made in this context has the same absolute information, but only some (or none) of it may be new, i.e. not yet in our database. Carnap and Bar-Hillel define a concept of *excess* information content by $\mathrm{CONT}(x/d) = \mathrm{CONT}(d \wedge x) - \mathrm{CONT}(d)$. Here d is the data (which can be represented by a single proposition if our language has only finitely many sentence symbols) and the excess content of x relative to d is obtained by finding the set-theoretic complement of CONT(d) relative to $\mathrm{CONT}(d \wedge x) = \mathrm{CONT}(d) \cup \mathrm{CONT}(x)$. Clearly $\mathrm{CONT}(x/d) = \mathrm{CONT}(x)$ if x and d have no content-element in common, and CONT(x/d) is empty if and only if d semantically entails x. In fact, it is not hard to see that $\mathrm{CONT}(x/d) = \mathrm{CONT}(d \rightarrow x)$. By analogy, cont(x/d) may be defined to be $\mathrm{cont}(d \wedge x) - \mathrm{cont}(d)$, thus providing a numerical measure of excess content. It is easy to show that $\mathrm{cont}(x/d) = k/m$ where k is now the number of content-elements in CONT(x/d) and m is the total number of content-elements. So of course $\mathrm{cont}(x/d) = \mathrm{cont}(d \rightarrow x)$.

There is however another sense in which the presence of data may diminish the *effective* (as opposed to absolute) information content of a statement. It may be that the information (or some of it) conveyed by a statement is *incompatible* with the data in the database. Since we may assume that data is carefully scrutinised before being incorporated into the database, it follows that unless the new information is specifically intended for the purpose of correcting certain items in the database, it is important to detect the incompatibility. Indeed, we may wish to add to the database those items of information that are compatible with existing data while reserving for more careful scrutiny those items that are incompatible. The analysis of incompatibility that follows, while it occurs within the theory of semantic information, is due to the present authors and has been partially reported only in [4, 5].

Inconsistency is the form of incompatibility that most readily springs to mind, although, as we shall see, it is not the only form. How does inconsistency manifest itself if we view the data d as the conjunction of content-elements (i.e. as a set of complete clauses)? A statement x establishes the contradiction if x contains the clauses which form the complement of some subset of d, since then $d \wedge x$ is the set of all clauses. If, for example, d consists of the single complete clause $\{p,q\}$, then in order to establish a contradiction x would have to contain the clauses $\{p,\neg q\}$, $\{\neg p,q\}$, $\{\neg p,\neg q\}$. Intuitively, the clause $d = \{p,q\}$ asserts that either p is the case or q is the case or both. Now, turning to x, the two clauses $\{p,\neg q\}$ and $\{\neg p,\neg q\}$ together assert that, whatever the case may be with regard to p, q does not obtain. And the two clauses $\{\neg p,q\}$ and $\{\neg p,\neg q\}$ together assert that, whatever the case may be with regard to q, p does not obtain. Thus the three clauses $\{p,\neg q\}$, $\{\neg p,q\}$ and $\{\neg p,\neg q\}$ jointly eliminate the possibilities held forth by $\{p,q\}$.

Weaker forms of incompatibility arise when the clauses of x eliminate (or cast doubt upon) only some of the possibilities held forth by the data d. For example, given $d = \{p,q\}$, the statement x consisting of the two clauses $\{p,\neg q\}$ and $\{\neg p,\neg q\}$ will not contradict d but will certainly eliminate the possibility that q obtains. Similarly, were the statement x to consist of the single clause $\{\neg p,\neg q\}$, no contradiction would be established nor would any one of the possibilities entertained by d be definitely eliminated, but doubt would be cast upon all those possibilities, since x says in effect "At least one of the possibilities entertained by d does not obtain". A yet weaker form of incompatibility would arise if the statement x were the single clause $\{p,\neg q\}$. Now x casts doubt upon, not all, but some of the possibilities entertained by d, namely the possibility that q obtains. Figure 2 summarises the discussion so far. The root node corresponds to the case $d = x = \{p,q\}$. Propositions are given in their most familiar forms.

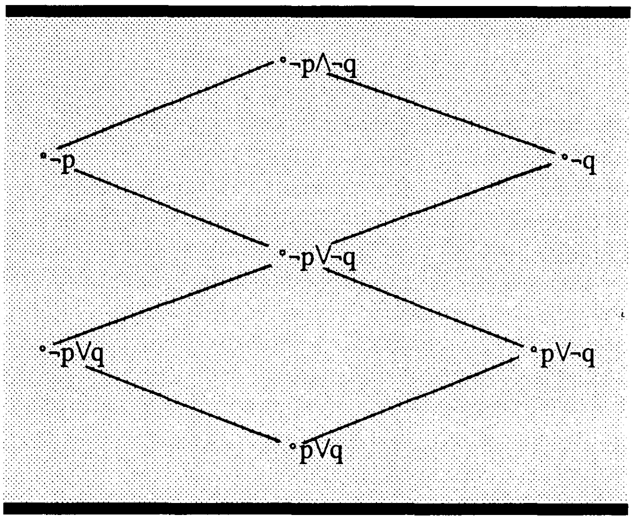

Figure 2

3. Automorphisms and Incompatibility

Consider the general case of a propositional language generated by the set of sentence symbols $S = \{p_1,p_2,\ldots,p_n\}$. As is well known, the set of propositions (or equivalence classes of strings modulo semantic equivalence) of such a language forms the free Boolean algebra F on n generators (the Lindenbaum algebra). Since F is finite (in fact, F has 2^m members with $m = 2^n$) it follows that F is atomic. Hence F is isomorphic to the power-set algebra C whose underlying set C is the power-set of the set of dual atoms of F, with intersection acting as join and union as meet. The isomorphism is given by the function CONT, since the dual atoms of F are precisely the content-elements of the language. We are now equipped to describe the most important property (from our point of view) of automorphisms of F, namely that the image of a proposition x under an automorphism contains exactly as much information as does x itself.

Theorem 2:
If $G : F \rightarrow F$ is any automorphism then G *preserves levels*, that is, for every $x \in F$, cont(x) = cont(G(x)).

Proof:
Assume G is an automorphism of F. Since G preserves order, dual atoms must go to dual atoms. Therefore CONT(x) and CONT(G(x)) have the same cardinality. ■

Having in mind the discussion of incompatibility summarised by figure 2, we now restrict our attention to the subclass of *literal automorphisms*, that is, automorphisms $G : F \rightarrow F$ for each of which a subset X_G of S exists such that, for every $x \in S$, $G(x) = \neg x$ if $x \in X_G$ and $G(x) = x$ otherwise. It is easy to show the following:

- Every literal automorphism is an involution, i.e. $G(G(x)) = x$ for every $x \in F$.
- There are precisely 2^n literal automorphisms which collectively form a Boolean algebra L with the identity map as its bottom and as its top the map * whose restriction to $\{p_1,...,p_n\}$ coincides with negation. The natural partial order $\leq$ on L places F before G if, for every $x \in S$, $G(x) = \neg x$ whenever $F(x) = \neg x$, that is, if $X_F \subseteq X_G$.

In the special case of the language generated from S = {p,q}, there are four literal automorphisms, corresponding to the four subsets of S. Let id denote the identity map, F_p the map which replaces each occurrence of the literal p by the literal ¬p, F_q the analogous map involving q, and * the map which replaces both p and q by their negations. Returning to figure 2, we note that the root node p∨q may be viewed as id(d) = id(p∨q), the nodes immediately above as $F_p(d)$ and $F_q(d)$ respectively, and the central node as d*. This suggests the following definitions.

Define an *incompatibility* to be any subset I of the Boolean algebra of literal automorphisms which is upwards hereditary, that is, which is such that if $F \in I$ and $F \leq G$, then $G \in I$. A content-element x is now said to be *I-incompatible* with a content-element y iff x = F(y) (equivalently F(x) = y) for some $F \in I$. More generally, x is I-incompatible with a proposition d iff x is I-incompatible with some $y \in$ CONT(d). The set of all content-elements I-incompatible with d is CONT(I(d)) = $\{x| \; x = F(y)$ for some $F \in I$ and some $y \in$ CONT(d)$\}$. I(d) may be seen as that proposition of which CONT(I(d)) is the complete clausal form. Hence we define the *I-compatible information content of x relative to the data d* to be the set *CONT(x) - CONT(I(d))*. This allows a numerical measure of compatible content to be defined in an obvious way: the *I-informativity* of x relative to d is given by
$cont_{I,d}(x) = k/i$ where k is the cardinality of CONT(x) - CONT(I(d)) and i is the number of content-elements I-compatible with d, that is, i is the cardinality of the complement of CONT(I(d)). Table 3 illustrates the values for the language generated by p and q, given d = p∨q and the incompatibilities I = $\{F_q, *\}$ and J = {*}.

x	$cont_{\{F_q,*\},p\vee q}(x)$	$cont_{\{*\},p\vee q}(x)$
p∨¬p	0	0
p∨q	1/4	1/4
q→p	0	1/4
p→q	1/4	1/4
¬p∨¬q	0	0
p	1/4	1/2
q	1/2	1/2
p+q	1/4	1/4
p↔q	1/4	1/2
¬q	0	1/4
¬p	1/4	1/4
p∧q	1/2	3/4
p∧¬q	1/4	1/2
¬p∧q	1/2	1/2
¬p∧¬q	1/4	1/2
p∧¬p	1/2	3/4

Table 3

The above definitions were suggested by the lower rhombus of figure 2. In particular, incompatibilities were defined in terms of upwards hereditary sets rather than single automorphisms because higher nodes cast doubt upon (or eliminate) more of the possibilities envisaged by the data than do lower nodes. What about the nodes above the lower rhombus? These all represent propositions that contain more than one clause each of which is I-incompatible with the data d = {p,q} for some I. For instance, ¬p entails the clauses {¬p,q} and {¬p,¬q}, so by virtue of the former clause ¬p is I-incompatible with d for any I such that $F_p \in I$, and by virtue of the latter clause ¬p is also I-compatible with d for any I such that $* \in I$. This suggests that the literal automorphisms suffice for an analysis of incompatibility.

4. Orderings, Probabilities and Informativity

Recall that every Boolean algebra has a natural partial order such that $x \leq y$ iff $x \wedge y = x$. In the case of the Lindenbaum algebra F, it so happens that $x \leq y$ iff x semantically entails y. Now in terms of absolute information content, we may say that y carries no more information than x iff $CONT(y) \subseteq CONT(x)$. The relation $\subseteq$ is the dual of the natural ordering on the Boolean algebra C, so by the isomorphism between C and F we may regard the ordering dual to $\leq$ on F, which we shall denote by $\sqsubseteq$ rather than by $\geq$, as reflecting the intuitive notion "carries no more information than". This ordering is in general not linear; there exist non-comparable elements. The quantitative measure cont does permit the comparison of arbitrary elements. Of course, the comparisons involving cont should not conflict with those of $\sqsubseteq$ where the latter exist. Fortunately, the relationship between CONT and cont ensures that *cont preserves* $\sqsubseteq$, that is, if $y \sqsubseteq x$ then $cont(y) \leq cont(x)$. Hence we may view cont as linearising $\sqsubseteq$. The measures $cont_{I,d}$ also linearise certain orderings that are not themselves in general linear.

Suppose we have data d, and denote CONT(I(d)) by D, where I is any fixed incompatibility. Hence D is the set of all content-elements I-incompatible with the data d. Recall that C is the set of all content-elements (i.e. dual atoms). Denote by D the Boolean algebra consisting of all subsets of C-D, with intersection as join and union as meet. There is a natural surjective homomorphism $G_d : C \rightarrow D$ given by $G_d(X) = X\text{-}D$. So the composition $G_d \circ CONT$ is surjective and has the significance of associating with a proposition x the I-compatible information content of x relative to the data d. Since $G_d \circ CONT$ is surjective, it induces a congruence $\approx_d$ on F such that $x \approx_d y$ iff $G_d(CONT(x)) = G_d(CONT(y))$.

Consider the quotient algebra $F/\approx_d$ whose members are the equivalence classes [x] of $\approx_d$. The quotient, being a Boolean algebra, has a natural ordering $\leq_d$ where $[x] \leq_d [y]$ iff $[x] = [x \wedge y]$. The dual ordering $\sqsubseteq_d$ is given by $[x] \sqsubseteq_d [y]$ iff $[x] = [x \vee y]$. The significance of the dual ordering is that $[x] \sqsubseteq_d [y]$ iff $G_d(CONT(x)) \subseteq G_d(CONT(y))$, so that $\sqsubseteq_d$ reflects the notion "carries no more I-compatible information than".

Theorem 3:
Given data d and a fixed incompatibility I, the I-informativity function $cont_{I,d}(x) : F \rightarrow [0,1]$ can be factored as the composition $J \circ H$, where $H : F \rightarrow F/\approx_d$ is the natural surjection given by $H(x) = [x]$ and $J : F/\approx_d \rightarrow [0,1]$ is given by $J[x] = cont_{I,d}(x)$. Furthermore, J preserves $\sqsubseteq_d$.

Proof:
J is well-defined, for if $x \approx_d y$ then $G_d(CONT(x)) = G_d(CONT(y))$ so that $CONT(x)\text{-}D = CONT(y)\text{-}D$ and therefore $cont_{I,d}(x) = cont_{I,d}(y)$.

To see that J preserves $\sqsubseteq_d$, assume $[x] \sqsubseteq_d [y]$.
Then $[x] = [x \vee y]$
i.e. $G_d(CONT(x)) = G_d(CONT(x \vee y))$
i.e. $CONT(x)\text{-}D = (CONT(x)\text{-}D) \cap (CONT(y)\text{-}D)$
i.e. $CONT(x)\text{-}D \subseteq CONT(y)\text{-}D$
i.e. $cont_{I,d}(x) \leq cont_{I,d}(y)$
so $J[x] \leq J[y]$. ■

We have shown that, like cont, the measures $cont_{I,d}$ preserve orderings that reflect information content of one sort or another. Another feature of cont is the connection with probability established by theorem 1. It is interesting to enquire after the relationship, if any, between the informativity measures $cont_{I,d}$ and probability. Some preliminary results for the case I = {*} follow. These results depend on the fact that when I contains a single automorphism, then the number of content-elements compatible with the data is equal to the number of valuations satisfying the data. These results will not in general hold for incompatibilities other than I = {*}; as * is the top of the algebra of literal automorphisms, all upwards hereditary subsets of L other than {*} contain two or more elements. In the proofs below we will indicate the cardinality of a set X by #X.

Theorem 4:
For all propositions x and d

$$cont_{*,d}(x) = 1 - prob_d(x^*)$$

where $prob_d$ denotes the obvious conditional probability measure derived from prob.

Proof:
Recall that $prob(x^*)$ can be calculated by counting the number of valuations that make x^* true and dividing by the total number of valuations (namely $m = 2^n$). Hence $prob_d(x^*)$ can be calculated by counting the number of valuations that satisfy both x^* and d, and then dividing by the number of valuations that satisfy d. In order to establish the theorem, we need to relate content-elements and valuations, so that, for example, the denominators in $cont_{*,d}(x)$ and $prob_d(x^*)$ can be shown to coincide.

Every valuation can be represented by the conjunction of a maximal consistent set of literals. Every content-element may be viewed as the negation of such a conjunction (courtesy of De Morgan). Therefore the number of valuations that satisfy x, say, is equal to the number of content-elements that do not appear in the complete clausal form of x. Furthermore, given that I contains the single automorphism *, the number of content-elements I-compatible with x is equal to the number of content-elements not entailed by x, which in turn is equal to the number of valuations satisfying x.
Now $1 - prob_d(x^*) = 1 - r/i = (i\text{-}r)/i$
where r is the number of valuations satisfying $d \wedge x^*$ and i is the number of valuations satisfying d.
Hence $r = \#[C - CONT(d^* \wedge x)]$
$= \#[C - (CONT(d^*) \cup CONT(x))]$
and $i = \#[C - CONT(d^*)]$.
Employing the simple identity $X\text{-}Y = Y'\text{-}(X \cup Y)'$, we

have that

$$\text{i-r} = \#[\mathrm{CONT}(x)-\mathrm{CONT}(d^*)]$$

and therefore $(i-r)/i = \mathrm{cont}_{*,d}(x)$. ■

Theorem 5:
For all propositions x and d

$$\mathrm{cont}_{*,d}(x) = \mathrm{cont}_{*,d^*}(x^*).$$

Proof:
Recall that $\mathrm{cont}_{*,d}(x) = k/i$,
where $k = \#G_d(\mathrm{CONT}(x))$
$\quad = \#[\mathrm{CONT}(x) - \mathrm{CONT}(d^*)]$
and $i = \#[C - \mathrm{CONT}(d^*)]$.
But $\#\mathrm{CONT}(x) = \#\mathrm{CONT}(x^*)$
and $\#\mathrm{CONT}(d^*) = \#\mathrm{CONT}(d)$.
Moreover, for every content-element c,
$c \in \mathrm{CONT}(x) \cap \mathrm{CONT}(d^*)$ iff $c^* \in \mathrm{CONT}(x^*) \cap \mathrm{CONT}(d)$.
Hence
$\#[\mathrm{CONT}(x)-\mathrm{CONT}(d^*)] = \#[\mathrm{CONT}(x^*)-\mathrm{CONT}(d)]$
and $\#[C - \mathrm{CONT}(d^*)] = \#[C - \mathrm{CONT}(d)]$.
So $k/i = \mathrm{cont}_{*,d^*}(x^*)$. ■

Corollaries:
For all propositions x and d

- $\mathrm{cont}_{*,d^*}(x) = \mathrm{cont}_{*,d}(x^*)$
- $\mathrm{prob}_d(x) = 1 - \mathrm{cont}_{*,d}(x^*)$
  $= 1 - \mathrm{cont}_{*,d^*}(x)$
  $= \mathrm{prob}_{d^*}(x^*)$
- $\mathrm{cont}_{*,d}(\neg x) = 1 - \mathrm{cont}_{*,d}(x)$.

5. The Infinite Case

Ultimately one would wish to generalise the theory of semantic information to apply to predicate logic (or some useful subset thereof that permits the use of free variables). At present it is not clear how to achieve this goal. A first step might be to treat propositional languages generated by an infinite set of sentence symbols. However, Brink and Heidema in [1] show that content-elements do not exist in such languages. Complete clauses obviously do not exist when there are infinitely many sentence symbols. What is demonstrated in [1] is the more interesting fact that such a language contains no propositions that may reasonably be substituted for complete clauses in our definition of semantic content. There are no logically weakest non-tautological propositions; the Lindenbaum algebra has no dual atoms. A possible line of attack, preserving the spirit of the semantic approach if not its precise form, would use valuations. In the finite case, a content-element (or complete clause) can readily be seen to be a proposition which is assigned the value T by all except a single valuation. Hence the cardinality of CONT(x) indicates the size of the subset of valuations that do not satisfy x. Accordingly we may choose, in the infinite case, to *define* CONT(x) to be the set of valuations that assign x the value F.

References

[1] C Brink and J Heidema, [1989], A verisimilar ordering of propositional theories: the infinite case, *Technical Report TR-ARP-1/89*, Automated Reasoning Project, Australian National University, Canberra.

[2] R Carnap, [1950], *Logical foundations of probability*, University of Chicago Press.

[3] R Carnap and Y Bar-Hillel, [1952], An outline of a theory of semantic information, *Technical report No. 247,* Research Laboratory of Electronics, Massachusetts Institute of Technology. Reprinted as chapter 15 of Y Bar-Hillel, [1964], *Language and Information,* Addison-Wesley.

[4] W A Labuschagne and J Heidema, [1988], Semantic information and evidence, *Verslagreeks van die Departement Wiskunde, RAU, 1/88.*

[5] W A Labuschagne and J Heidema, [1989], The measurement of semantic information, in G Lasker (ed), *Advances in Computer Science*, International Institute for Advanced Studies in Systems Research and Cybernetics, University of Windsor, Canada.

[6] G Rzevski, [1985], On criteria for assessing an information theory, *The Computer Journal,* **28**(3), 200-202.