

A Formal Logical Framework for Cadiag-2

Agata CIABATTONI^a, Thomas VETTERLEIN^{b,1}, Klaus-Peter ADLASSNIG^b

^a*Institute for Discrete Mathematics and Geometry,*

Vienna University of Technology, Austria

^b*Section on Medical Expert and Knowledge-Based Systems,*

Medical University of Vienna, Austria

Abstract. Cadiag-2—where “Cadiag” stands for “computer-assisted diagnosis”—is an expert system based on fuzzy logic assisting in the differential diagnosis in internal medicine. With its aid, it is possible to derive from possibly vague information about a patient’s symptoms, signs, laboratory test results, and clinical findings conjectures about present diseases. In this paper, we provide a mathematical formalization of the inferential mechanism of Cadiag-2. The aim is to have a formal logical calculus at hand which corresponds to the mode of operation of Cadiag-2 and which is among others needed to perform consistency checking of Cadiag-2’s medical knowledge base.

Keywords. medical expert systems, Cadiag-2, fuzzy logic, logical calculus, internal medicine

1. Introduction

Cadiag-1 and Cadiag-2 are computer-based medical consultation systems, developed at the University of Vienna Medical School (now Medical University of Vienna) since the 1980’s (see, e.g., [1]; for the performance, see, e.g., [2]). Their aim is to support clinical differential diagnosis in the field of internal medicine. Both systems are based on relationships between propositions about symptoms, signs, laboratory test results, and clinical findings (symptoms for short) on the one hand and diagnoses on the other hand.

In Cadiag-1, these propositions are treated as three-valued, that is, as being true, false, or undefined. The relationships between these propositions can be formulated in the monadic fragment of first-order classical logic; the decidability of the latter makes it possible to check the consistency of Cadiag-1’s medical knowledge base; and actually 17 inconsistencies within 50,000 binary relationships have once been detected [3].

Precise information, however, is often not available to physicians to decide about a patient’s disease. In order to process vague information, the successor system Cadiag-2 was based on fuzzy logic [4].

Note that Cadiag-2 relies entirely on fuzzy techniques. An advantage of this choice, when compared to systems like DXplain [5], which are essentially based on probability theory, is the fact that Cadiag-2 inferences are always offered together with a justification which is easily comprehensible to the user.

¹ Corresponding Author: Spitalgasse 23, A-1090 Vienna; E-Mail: thomas.vetterlein@meduniwien.ac.at.

However, Cadiag-2 has not been explicitly formulated in the framework of a formal logic. As a consequence, the problem of how to check the consistency of its rules is not yet well understood. In this paper, we provide first steps towards solving this issue; we introduce a basic fuzzy-logical framework for Cadiag-2.

2. The Cadiag-2 Inferential Mechanism

We shall shortly describe the inferential mechanism of Cadiag-2. For a comprehensive description of the system, see, e.g., [6]. The knowledge base of Cadiag-2 consists of if-then rules representing definitional, causal, statistical, and heuristic one-to-one or many-to-one relationships between symptoms and diseases. On the basis of this general knowledge and the particular information referring to a patient (recounted history, observed signs, measured test results), the inference engine can draw conclusions. We note that symptoms and diseases are not analysed with respect to their meaning, but are rather treated as pure propositions; what matters are their mutual relationships.

Propositions processed by Cadiag-2. An example of a proposition referring to a symptom might be “suffering from strong abdominal pain”. It is obvious that the alternatives *true* and *false* to evaluate this proposition are not exhaustive. Accordingly, Cadiag-2 considers the statements about symptoms as being vague. Namely, to each symptom, there is associated a *degree of presence*, expressed by any element of the real unit interval $[0,1]$.

The second class of propositions in Cadiag-2 refers to diagnoses. It is often not or not yet possible to confirm or to exclude a diagnosis with certainty, thus, to each diagnosis, there is associated a *degree of certainty*, which is again a value in $[0,1]$.

Let now $\sigma_1, \dots, \sigma_m$ be all symptoms and $\delta_1, \dots, \delta_n$ all diagnoses contained in Cadiag-2's knowledge base. Each such symbol is called a *basic entity*. By the use of *connectives*, we can form *compound entities*; we have to our disposal conjunction \wedge , disjunction \vee , and negation \sim .

For example, $\sigma_1 \wedge \sim \sigma_3$ expresses the presence of the symptom σ_1 and the absence of the symptom σ_3 . Assume now that σ_1 and σ_3 are assigned the truth values t_1 and t_3 , respectively. Then we may calculate a truth value for $\sigma_1 \wedge \sim \sigma_3$ as well, namely, we take $\min\{t_1, 1-t_3\}$. In general, if we are given an assignment of certain basic entities, we may extend it to as many compound ones as possible:

Definition 1. An *evaluation* is a function v from a subset of the set of entities to the real unit interval $[0,1]$ such that the following holds: (i) If $v(\alpha) = s$ and $v(\beta) = t$, then $v(\alpha \wedge \beta) = \min\{s, t\}$ and $v(\alpha \vee \beta) = \max\{s, t\}$; (ii) if $v(\alpha) = 0$ or $v(\beta) = 0$, then $v(\alpha \wedge \beta) = 0$; (iii) if $v(\alpha) = t > 0$ and $v(\beta)$ is undefined, or $v(\alpha)$ is undefined and $v(\beta) = t > 0$, then $v(\alpha \vee \beta) = t$; (iv) if $v(\alpha) = t$, then $v(\sim \alpha) = 1 - t$.

The input of one run of Cadiag-2 is an evaluation w_0 , called the *initial evaluation* and used to describe the state of a particular patient. Then, inference rules are successively applied so as to generate a sequence of evaluations w_1, w_2, \dots . Compared

to its predecessor, each evaluation in this sequence encodes an increased amount of information about the patient. The process terminates after finitely many, say l , steps, and w_l is called the *final evaluation*.

The rules. For each $k = 1, \dots, l$, w_k is the result of an application of a *rule* to w_{k-1} . Each rule, say R , originates from the knowledge base of Cadiag-2 and contains the following information: (i) a possibly compound entity α , (ii) a basic entity β , and (iii) the *type* of the logical relationship between α and β , which is one of the following:

- (c_d), where $d \in (0,1]$. Then R expresses that α , which encodes, e.g., a combination of symptoms, gives a hint to β , which in turn encodes, e.g., a diagnosis. The implication holds the stronger the larger d is; d is called the *confirmability degree*.
- (me) Then R expresses that α and β are mutually exclusive.
- (ao) Then R expresses that if β holds, then necessarily also α holds.

Let R be of type (c_d), relating the entities α and β . Then R is applied to w_{k-1} as follows. The truth value t assigned to α and the confirmability degree d are combined to one truth value $b = \min\{t, d\}$. If then β is not yet in the domain of w_{k-1} , we put $w_k(\beta) = b$. If otherwise $w_{k-1}(\beta) > 0$ and $b > 0$, we put $w_k(\beta) = \max\{w_{k-1}(\beta), b\}$. If $w_{k-1}(\beta) = 0$ and $b < 1$ or if $w_{k-1}(\beta) < 1$ and $b = 0$, then $w_k(\beta) = 0$. For the remaining basic entities, w_{k-1} coincides with w_k , and the compound entities are defined according to Definition 1.

Consider the following example of a rule of type (c_{0.30}):

<p>IF <i>suspicion of liver metastases by liver palpation</i> THEN <i>pancreatic cancer</i> with the confirmability degree 0.30.</p> <p>If, say, there is a clear suspicion of liver metastases by palpation, we evaluate the assumption of this rule with 1. An application of the rule then associates to the conclusion, unless there is better information available, a certainty degree 0.30.</p>

The cases (me) and (ao) work similarly.

The rules are applied systematically one by one, but the order is arbitrary. The process is completed if, by use of any of the rules, the evaluation remains unchanged.

3. CadL—the Logical Counterpart of Cadiag-2

In this section, we introduce CadL (“Cadiag logic”), a calculus adequate to formalise Cadiag-2. According to the ideas of Cadiag-2, CadL uses a concept well-known in fuzzy logic (see, e.g., [7]): pairs consisting of a proposition and a rational truth value.

Definition 2. The *atomic propositions* of CadL are symbols $\varphi_1, \varphi_2 \dots$. The *lattice propositions* of CadL, denoted by F_L , are the expressions built up from the

atomic propositions by means of the connectives \wedge , \vee , and \sim . Moreover, the implications of **CadL**, denoted by F_I are the expressions $\alpha \rightarrow \beta$, where $\alpha, \beta \in F_L$. Finally, $F_L \cup F_I$ are the propositions of **CadL**.

A graded proposition is a pair (φ, t) , where $\varphi \in F$ and $t \in [0, 1]$.

An entity in Cadiag-2 together with its image under an evaluation, corresponds to a graded proposition in **CadL**.

Definition 3. The evaluation rules of **CadL** are

$$\frac{(\alpha, s) \quad (\beta, t)}{(\alpha \wedge \beta, s \wedge t)} \quad \frac{(\alpha, 0)}{(\alpha \wedge \beta, 0)} \quad \frac{(\beta, 0)}{(\alpha \wedge \beta, 0)} \quad \frac{(\alpha, t)}{(\sim \alpha, \sim t)}$$

$$\frac{(\alpha, s) \quad (\beta, t)}{(\alpha \vee \beta, s \vee t)} \quad \frac{(\alpha, u)}{(\alpha \vee \beta, u)} \quad \frac{(\beta, u)}{(\alpha \vee \beta, u)}$$

for any $\alpha, \beta \in F_L$ and $s, t \in [0, 1], u \in (0, 1]$.

The manipulation rules are

$$\frac{(\alpha \rightarrow \beta, d) \quad (\alpha, t)}{(\beta, d \wedge t)} \quad \text{for any } d, t > 0$$

$$\frac{(\alpha \rightarrow \sim \beta, 1) \quad (\alpha, 1)}{(\beta, 0)} \quad \frac{(\alpha \rightarrow \sim \beta, 1) \quad (\alpha, 0)}{(\beta, 0)}$$

for any $\alpha, \beta \in F_L$ such that β is atomic.

A theory of **CadL** is a finite set T of graded propositions. A proof from T is a finite sequence of graded propositions each of which is either in T or the conclusion of a rule whose assumptions are among the preceding elements of the proof. The last entry in a proof from T is called *provable* from T .

The evaluation rules serve to determine the truth values associated to compound propositions; and the three manipulation rules mirror the three types of rules of Cadiag-2.

We will now establish the correspondence between Cadiag-2 and **CadL**. Given a Cadiag-2 knowledge base, we identify each basic entity with a unique atomic proposition of **CadL**, and each compound entity with the respective lattice proposition of **CadL**. Let us fix some initial evaluation w_0 of a run of Cadiag-2. We associate with w_0 the following theory T_{w_0} of **CadL**: (i) $(\varphi, w_0(\varphi))$ if $\varphi \in F_A$ is in the domain of w_0 ; (ii) $(\alpha \rightarrow \beta, d)$ for each rule in the knowledge base of type (c_d), where $d \in (0, 1]$; (iii) $(\alpha \rightarrow \sim \beta, 1)$ for each rule in the knowledge base of type (me); (iv) $(\sim \alpha \rightarrow \sim \beta, 1)$ for each rule in the knowledge base of type (ao).

Proposition 1 (completeness). Let β be an entity in the domain of the final evaluation w_1 of a run of Cadiag-2. Then, $(\beta, w_1(\beta))$ is provable in **CadL** from T_{w_0} .

The converse direction is more delicate as not all the proofs in **CadL** correspond to a run of Cadiag-2. The reason is that when a new value is computed at the k -th step of

a run of Cadiag-2, the previously obtained value for the same entity may become obsolete. We strengthen the notion of a proof in CadL.

Definition 4. Call a proof of CadL *strict* if the following holds. Let the i -th proof entry be derived by a rule, and let the j -th entry be among its assumptions, being of the form (α, t) for some $\alpha \in F_L$ and $t \in (0, 1]$. Then neither of the entries $j+1, \dots, i-1$ is of the form (β, u) , where β is a subformula of α ; and neither of the entries prior to i is (α, t') for some $t' > t$ or $t' = 0$.

Proposition 2 (soundness). *Let P be a strict proof of CadL from T_{w_0} , and let (β, t) be contained in P , where $\beta \in F_L$. Then there is a run of Cadiag-2 with l steps such that $w_{l'}(\beta) = t$ for some $l' \leq l$.*

Propositions 1 and 2 together imply that (initial pieces of) runs of Cadiag-2 and strict proofs of CadL are in an exact mutual correspondence.

4. Discussion and Conclusion

We have shown that the mode of operation of Cadiag-2 can be represented in the framework of a formal logical calculus, called CadL. Any general question about the inference of Cadiag-2 translates to a question about this logic.

Moreover, we are able to characterize the CadL system within the family of t -norm-based fuzzy logics, which are studied intensively. We have furthermore prepared the ground for tackling one of the most important problems about the Cadiag-2 knowledge base, its consistency.

Acknowledgement. This research was partially supported by the Vienna Science and Technology Fund (WWTF) Grant MA07-016.

References

- [1] Adlassnig, K.P., Kolarz, G. (1982) Cadiag-2: Computer-assisted medical diagnosis using fuzzy subsets. In Gupta, M.M., Sanchez, E. (eds.) *Approximate Reasoning in Decision Analysis*, North-Holland Publishing Company, Amsterdam, 219–247.
- [2] Leitich, H., Adlassnig, K.-P., Kolarz, G. (2002) Evaluation of two different models of semiautomatic knowledge acquisition for the medical consultant system CADIAG-II/RHEUMA. *Artificial Intelligence in Medicine* 25(3):215–225.
- [3] Moser, W., Adlassnig, K.-P. (1992) Consistency checking of binary categorical relationships in a medical knowledge base. *Artificial Intelligence in Medicine* 4(5):389–407.
- [4] Zadeh, L.A. (1965) Fuzzy sets. *Information and Control* 8(3):338–353.
- [5] Barnett, G.O., Cimino, J.J., Hupp, J.A., Hoffer, E.P. (1987) DXplain, An Evolving Diagnostic Decision-Support System. *Journal of the American Medical Association* 258(1):67–74.
- [6] Adlassnig, K.-P., Kolarz, G., Scheithauer, W., Grabner, H. (1986) Approach to a hospital-based application of a medical expert system. *Medical Informatics* 11, 205–223.
- [7] Hájek, P. (1998) *Metamathematics of Fuzzy Logic*. Kluwer, Dordrecht.