Identification of Semantics: Balancing between Complexity and Validity

Manolis Falelakis, Christos Diou and Anastasios Delopoulos
Department of Electrical and Computer Engineering,
Aristotle University of Thessaloniki - Greece
Email: {manf, diou}@olympus.ee.auth.gr, adelo@eng.auth.gr

***Abstract*— This paper addresses the problem of identifying Semantic Entities (e.g., events, objects, concepts etc.) in a particular environment (e.g., a multimedia document, a scene, a signal etc.) by means of an appropriately modelled *Semantic Encyclopedia*. Each Semantic Entity in the encyclopedia is defined in terms of other Semantic Entities as well as low level features, which we call *Syntactic Entities*, in a hierarchical scheme. Furthermore, a methodology is introduced, which can be used to evaluate the direct contribution of every syntactic feature of the document to the identification of Semantic Entities. This information allows us to estimate the quality of the result as well as the required computational cost of the search procedure and to balance between them. Our approach could be particularly important in real time and/or bulky search/indexing applications.**

I. INTRODUCTION

SEMANTIC analysis of data tends to become a necessity in modern multimedia applications coping with the need to organize multimedia documents and provide a higher level of interaction between humans and computers. It is against this background that researchers have recently put great effort on developing semantic extraction algorithms and frameworks for standardizing semantic descriptions, such as MPEG-7. Among the issues that arise is the control of computational complexity associated with such procedures, a need which becomes more apparent in time critical applications.

In this work we propose a methodology which can act as a complexity controlling mechanism by designing efficient methods for identifying Semantic Entities, taking into account the tradeoff between limitations of computational cost (i.e. algorithmic complexity) versus obtained validity of the result. Our formulation is based on the notion of the Semantic Encyclopedia that allows for description of Semantic Entities based on other Semantic and/or Syntactic Entities. More specifically, we assume that existence of a Syntactic/Semantic Entity implies, *in a certain degree*, existence of a higher level Semantic Entity.

In the next section the structure of the Semantic Encyclopedia is presented. Section III displays a means of directly linking Semantic Entities with Syntactic Features allowing for the definition of identification metrics that are introduced in section IV, as well as for designing the search procedure, as presented in section V. Experimental results displaying the value of our approach are included in section VI. Finally, concluding remarks and open issues are found in the last section.

II. SEMANTIC ENCYCLOPEDIA - DEFINITIONS

A. Syntactic Entities

As *syntactic feature* t we define any measurable quantity (eg. brightness, frequency, straightness etc) that can be obtained by applying a *corresponding algorithm* on a given data set. For simplicity we assume real valued syntactic features, either 1-dimensional (eg. brightness on R) or multi-dimensional (eg. color on R^3).

A *Syntactic Entity* or property $y_i(t) \in [0, 1]$ is a fuzzy set on a syntactic feature t. For instance the property ”very bright” is defined on the feature ”brightness” as depicted in Fig. 1. We assign the label Y_i to a particular Syntactic Entity $y_i(t)$ and assume a finite set $\mathbf{Y} = \{Y_i\}$ of such labels corresponding to the entire collection of Syntactic Entities of interest.

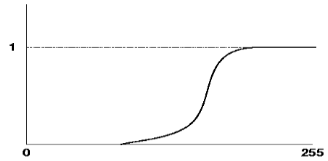

Fig. 1. Definition of property ”Very Bright”.

It is essential to point out that the aforementioned computational cost of the search procedure refers to the algorithms τ employed for measuring the data set under examination. Let t_τ be a feature evaluated employing the algorithm τ, then τ is used to assess the degree $\mu_{Y_i} \equiv y_i(t_\tau)$ up to which the particular data set assumes property Y_i,

B. Semantic Entities

Objects, events, categories or other concepts that may be handled by human perception/logic are collectively assigned the term *Semantic Entity*. In our discussion we assume a set $\mathbf{E}$ of Semantic Entities of interest with a further assumption that each entity with label $E_i \in \mathbf{E}$ can be ”described” on the basis of other Semantic Entities within $\mathbf{E}$ and/or Syntactic

Entities within $\mathbf{Y}$ in a manner explained below, in Section II-C. Note that $\mathbf{E}$ and $\mathbf{Y}$ form the building blocks of the Semantic Encyclopedia.

C. *Definitions*

The presented qualitative description of a Semantic Entity on the basis of simpler entities can be enriched by more quantitative information regarding the degree of relation between a Semantic Entity and its successors.

An entity E_k can be described by more than one alternative descriptions, each one providing different amount of information about it. We define as *reliability* m_{kJ} of a description J of E_k a real number in [0,1] measuring the amount and the quality of the information provided. Equivalently m_{kJ} is the degree up to which the particular description characterizes E_k.

In addition, Entities (either Syntactic or Semantic) that are included in a description have different importance quantified by a set of corresponding weights. These weights can be considered as elements of a fuzzy relation on $\mathbf{S} \times \mathbf{S}$, where $\mathbf{S} \equiv \mathbf{Y} \bigcup \mathbf{E}$, is a set containing all Semantic and Syntactic Entities of the encyclopedia (see [1] for a similar discussion). For a particular Semantic Entity $E_k \in \mathbf{E}$ as *relevance* we define $F_{kJ} : \mathbf{S} - \{E_k\} \rightarrow [0,1]$ for those $S_i \in \mathbf{S}$, participating in a certain description J (one of possible alternatives). We call *definition* of E_k in terms of J the discrete fuzzy set

$$E_{kJ} = F_{kJ1}/S_1 + F_{kJ2}/S_2 + ... + F_{kJn}/S_n \tag{1}$$

In Eq. (1) we note that S_1 implies existence of E_{kJ} up to the degree F_{kJ1}. Definitions of this type are either included in the semantic encyclopedia (*primary definitions*) or are derived from a substitution procedure. By gradually substituting all Semantic Entities that appear in (1) we conclude to detailed definitions of the form of (2). A *detailed definition* is a definition that contains only Syntactic Entities, i.e.

$$E_{kJ_d} = F_{kJ_d1}/Y_1 + F_{kJ_d2}/Y_2 + ... + F_{kJ_dm}/Y_m . \tag{2}$$

By replacing the Semantic Entities with their respective descriptions and by repeating this procedure recursively, any primary definition can be transformed into a detailed one. Note that from a single primary definition of an entity, a multiplicity of alternative detailed definitions may be produced, since substituted entities may have alternative descriptions.

III. GENERATION OF DETAILED DEFINITIONS

When forming a detailed definition, we should ensure that the relevance factors that occur obtain appropriate values. A decomposition method has been devised for this purpose and is presented here with an example. Consider the definitions shown in Fig. 2, where capital, lowercase and indexed capital letters, denote Semantic Entities, Syntactic Entities and alternative definitions respectively. Entity $\mathcal{A}$ is defined by two alternative descriptions, while entity $\mathcal{C}$ has only one description. By substituting $\mathcal{C}$ in the description J_1 of $\mathcal{A}$ we come up with a new description J_d of $\mathcal{A}$ which depends on the Syntactic Entities a and b:

$$A_{J_d} \quad = \quad F_{AJ_da}/a + F_{AJ_db}/b \tag{3}$$

In order to determine sensible values for the relativity factors F, we use a fuzzy intersection operator (t-norm) $\mathcal{I}$ for the "transition" from A_{J_1} to b via $\mathcal{C}$, including the reliability of the definition C_{J_3}. Hence, $F_{AJ_db} = \mathcal{I}(F_{AJ_1C}, \mathcal{I}(m_{J_3C}, F_{CJ_3b}))$. The same procedure is not sufficient for the calculation of F_{AJ_da}, since A_{J_1} is related to a directly with F_{AJ_1a} and via C_{J_3} with $\mathcal{I}(F_{AJ_1C}, \mathcal{I}(m_{J_3C}, F_{CJ_3a}))$. We use a fuzzy union operator (t-conorm) $\mathcal{U}$ to combine the two values, consequently $F_{AJ_da} = \mathcal{U}(\mathcal{I}(F_{AJ_1C}, \mathcal{I}(m_{J_3C}, F_{CJ_3b})), F_{AJ_1a})$. By replacing those values in Eq. (3) one comes up with the definition A_{J_d} of $\mathcal{A}$. Information regarding fuzzy intersection and union operations can be found in [2].

It is important to mention that the procedure used for the calculation of the values of F can be different, as it does not affect the proposed method for quality and complexity control. However, it is essential to transform the primary definitions of the Semantic Encyclopedia into detailed ones (i.e. depending only on Syntactic Entities).

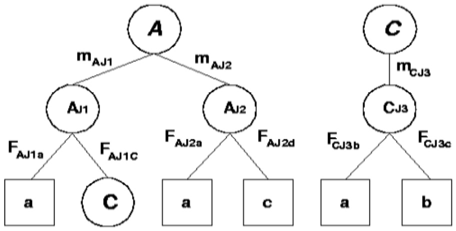

Fig. 2. A primary definition.

IV. VALIDITY, COMPLEXITY AND CERTAINTY

In order to quantify the information provided by a detailed definition and the computational cost required to evaluate it, we next introduce *Validity* and *Complexity* of the definition.

For a detailed definition $E_{kJ} = \sum_i F_{kJi}/Y_i$ we define Validity as

$$V_{kJ} \triangleq \mathcal{I}(m_{kJ}, (\underset{i}{\mathcal{U}}(F_{kJi})) \tag{4}$$

Complexity is defined as

$$C_{kJ} \triangleq \sum_i c(\tau_i) \tag{5}$$

where $c(\tau_i)$ denotes the computational cost of algorithm τ_i run to evaluate the presence of syntactic property Y_i.

In addition we define the metric of *Certainty* to quantify the degree of our belief that E_k (as defined by E_{kJ}) has been identified within a particular data set, as

$$\mu_{kJ} \triangleq \mathcal{I}(m_{kJ}, (\underset{i}{\mathcal{U}}(\mathcal{I}(\mu_{Y_i}, F_{kJi})))) \tag{6}$$

We must point out that Certainty depends on the results of identification algorithms τ_i and actually provides a metric regarding the identification of an entity in a specific multimedia document. On the other hand, Validity and Complexity are ”a priori” computable referring to the quality of the detailed definition and the computational cost it entails. Note also that Certainty cannot be greater than Validity ($\mu_{kJ} \leq V_{kJ}$), as the first one (Eq. (6)) includes an extra intersection with the results μ_{Y_i} of the evaluation of Syntactic Features.

V. SEMANTIC SEARCH DESIGN

In our formulation, having obtained a detailed definition, the order of evaluation of syntactic properties (equivalently the execution of required algorithms) is of no importance. Thus having evaluated only a subset of properties $\mathbf{A} \subseteq \mathbf{Y}^J = \{Y_1, \ldots, Y_m\}$ we can define partial Validity, Complexity and Certainty as $V_{kJ}(\mathbf{A}) \triangleq \mathcal{I}(m_{kJ}, (\underset{i\in\mathbf{A}}{\mathcal{U}}(F_{kJi})))$, $C_{kJ}(\mathbf{A}) \triangleq \sum_{i\in\mathbf{A}} c(\tau_i)$ and $\mu_{kJ}(\mathbf{A}) \triangleq \mathcal{I}(m_{kJ}, (\underset{i\in\mathbf{A}}{\mathcal{U}}(\mathcal{I}(\mu_{Y_i}, F_{kJi}))))$ respectively.

An efficient way of modelling the design process is by the use of an automaton. Each state of such automaton is labelled by an ordered pair $(\mathbf{A}, \mathbf{Y}^J - \mathbf{A})$, denoting the set of evaluated algorithms and the remaining ones respectively. The single final state, which corresponds to evaluation of all syntactic properties of $\mathbf{Y}^J$, is labelled by $(\mathbf{Y}^J, \emptyset)$ while the initial state is of the form $(\emptyset, \mathbf{Y}^J)$. The automaton that corresponds to definition of Fig. 2 is depicted in Fig. 3 where state q_3, for instance, is labelled with $(\{b, c, d\}, \{a\})$.

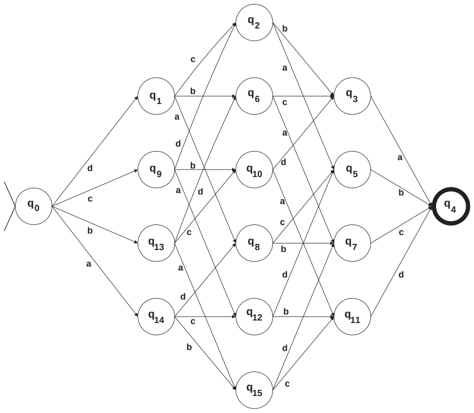

Fig. 3. Automaton modelling the search procedure.

We may observe that each state of the automaton can be assigned a corresponding partial Validity and Complexity which can characterize that state. Hence, the problem of designing an optimal search procedure under Validity and Complexity constraints, can be transformed to a problem of finding the optimal state of the automaton that satisfies these constraints. For more information on modelling with finite automata one can refer to [3], while general information on finite automata can be found in [4].

A. Design Methodologies

When Validity is of main concern in the search process, we set a Validity threshold M, under which no answer can be accepted. From all the states of the automaton that comply with the restriction $V_{kJ}(\mathbf{A}) \geq M$ we seek the state that requires minimum Complexity. Reversely, when searching under a specific Complexity budget $C > 0$ we first locate all states satisfying $C_{kJ}(\mathbf{A}) \leq C$ and then choose the one that maximizes Validity.

These simple methodologies can be enhanced by the use of more sophisticated search techniques. As an example, we could begin the search using a low Complexity threshold and to continue only should we receive satisfactory Certainty results. Such approach would prove to be useful in case there is a multiplicity of entities E_k,$(k = 1, \ldots, N)$ to be identified at real time with a limited Complexity budget.

VI. EXPERIMENTS

As an experiment the system was asked to identify the Entity ”table” E_{01}, defined as $E_{01} = 0.9/Y_{01} + 0.7/E_{02}$ where Y_{01} represents the Syntactic Entity ”horizontal surface” and E_{02} the Semantic Entity ”two legs” which in turn is defined as $E_{02} = 0.6/Y_{02} + 0.9/Y_{03} + 0.8/Y_{04}$. Y_{02}, Y_{03} and Y_{04} correspond to the Syntactic Entities ”two straight lines”, ”two vertical lines” and ”same length” respectively.

As a fuzzy intersection operator, the product was chosen: $\mathcal{I}(a, b) = ab$ and its complementary, the algebraic sum, for union: $\mathcal{U}(a, b) = a + b - ab$. Composing the two descriptions as described before, the following primary definition of ”table” was obtained:

$$E_{01} = 0.9/Y_{01} + 0.378/Y_{02} + 0.567/Y_{03} + 0.567/Y_{04} \quad (7)$$

Moreover, the algorithms used to evaluate the syntactic properties were assigned estimates of Complexity values as displayed in the following table.

Algorithm	Complexity
Horizontal surface	$\mathcal{C}(1) = 3.6$
Two straight lines	$\mathcal{C}(2) = 4.8$
Two vertical lines	$\mathcal{C}(3) = 4.5$
Two lines of same length	$\mathcal{C}(4) = 3.3$

As of this point it was possible to design the search process for various Validity and Complexity thresholds. Using the sample drawings shown in Fig. 4 which present ”versions” of a table, the semantic search was performed , calculating up to which degree (Certainty) each drawing represents the entity ”table”.

Results of design in terms of Validity are illustrated in the first four columns of Figure 5, while the corresponding attained Certainty values for each drawing (a)-(f) have been included in the next six columns. Rows of the table correspond to design setting Validity threshold to $M = 0.2, 0.46, 0.73, 0.785$. Two comments are worth to be made: (1)Modifying M results in selection of different algorithms

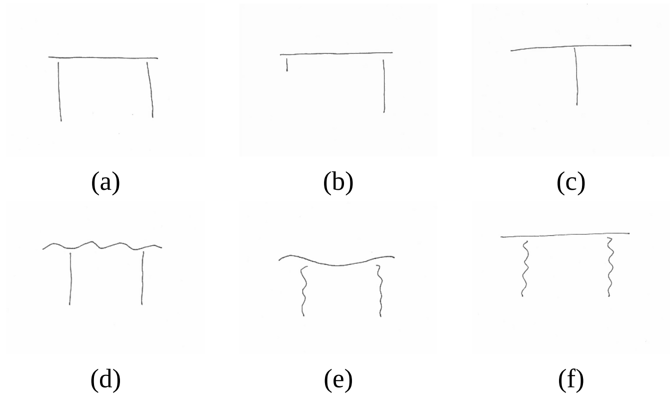

Fig. 4. Drawings of ”tables”

(see e.g. rows one and two). (2) Relatively high Validity and Certainty is obtained at reasonably low computational cost, but pushing the Validity threshold to its high levels causes abrupt increase of the required Complexity. Similarly, design results in terms of Complexity have been included in the table of Figure 6 for Complexity bounds $C = 3.7,\ 8,\ 13,\ 7$. Commenting on these results, decent Validity levels are attained even under low Complexity constraints. Allowing higher Complexity budgets enhances both Validity and Certainty but the gained increase is not proportional to the additional computational cost.

M	*Validity*	*Complexity*	*Algorithms*	*(a)*	*(b)*	*(c)*	*(d)*	*(e)*	*(f)*
0.2	0.45	3.3	4	0.4	0.09	0	0.438	0.432	0.403
0.46	0.72	3.6	1	0.705	0.7	0.675	0.396	0.188	0.685
0.73	0.77	6.9	1, 4	0.753	0.703	0.675	0.617	0.519	0.743
0.785	0.79	16.2	1, 2, 3, 4	0.782	0.763	0.675	0.73	0.64	0.765

Fig. 5. Design in terms of Validity.

C	*Validity*	*Complexity*	*Algorithms*	*(a)*	*(b)*	*(c)*	*(d)*	*(e)*	*(f)*
3.7	0.72	3.6	1	0.705	0.7	0.675	0.396	0.188	0.685
8	0.7654	6.9	1, 4	0.753	0.703	0.675	0.617	0.519	0.743
13	0.7827	11.4	1, 3, 4	0.771	0.746	0.675	0.692	0.572	0.752
17	0.79	16.2	1, 2, 3, 4	0.782	0.763	0.675	0.73	0.64	0.765

Fig. 6. Design in terms of Complexity.

To examine the behavior of the proposed method with a variety of algorithm complexity values and relativity factors, a set of synthetic experiments was conducted using descriptions with random values. Figures 7 and 8 display results, where the continuous line corresponds to uniform distribution of F in $[0, 1]$ and complexities in $[1, 10]$, while for the dotted line normal distribution of complexities was used, with a mean value of 5 and standard deviation 0.5. The remarks of the previous experiment are confirmed by these results. Furthermore, high efficiency of the methodology is observed when dealing with widely distributed values, whereas distributions with low deviation tend to decrease the quality of the result. One could expect this, since algorithms with similar values make the selection between them unimportant (they can be considered ”equivalent” under our scope) and vice versa.

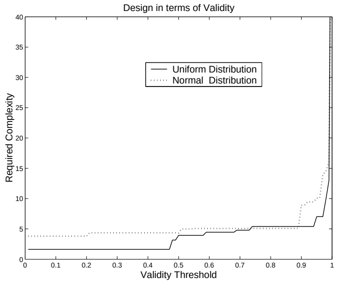

Fig. 7. Design in terms of Validity.

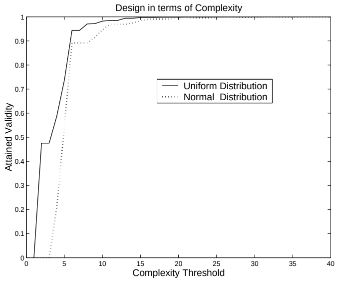

Fig. 8. Design in terms of Complexity.

VII. CONCLUSIONS

In this work, we presented a methodology which allows efficient semantic search in terms of quality (Validity), while limiting the required computational cost (Complexity) at sustainable levels. Experimental results showed that the proposed method can handle this tradeoff effectively and can prove to be very useful in time critical applications with limited complexity budgets.

Issues to be addressed are the enrichment of the encyclopedia by using proper mathematical logics (e.g. description logics) and different types of relations between entities (e.g. position-related). Another open issue is dealing with the exponential size of the automatons that occur, a problem which seems to be a variation of the Knapsack problem, only with a non-linear gain function.

REFERENCES

[1] G. Akrivas, G. B. Stamou and S. Kollias. Semantic Association of Multimedia Document Descriptions through Fuzzy Relational Algebra and Fuzzy Reasoning *IEEE Transactions On Systems, Man and Cybernetics, part A.*, **34** 2004

[2] George J. Klir and Bo Yuan. *Fuzzy Sets and Fuzzy Logic; Theory and Applications.* Prentice Hall, 1995

[3] P. Panagiotopoulos, M. Falelakis and A. Delopoulos. Efficient Semantic Search using Finite Automata. *Proceedings of the 6th COST276 Workshop, Thessaloniki, Greece*, 2004.

[4] Harry R. Lewis and Christos H. Papadimitriou. *Elements of the Theory of Computation.* Prentice Hall, 1998