

Attribute-Oriented Fuzzy Generalization in Proximity- and Similarity-Based Relational Database Systems

Rafal A. Angryk,^{1,*} Frederick E. Petry^{2,†}

¹Computer Science Department, Montana State University,
Bozeman, MT 59717-3880, USA

²EECS Department, Tulane University, New Orleans, LA 70118, USA

In this article we investigate an attribute-oriented induction approach for acquisition of abstract knowledge from data stored in a fuzzy database environment. We utilize a proximity-based fuzzy database schema as the medium carrying the original information, where lack of precise information about an entity can be reflected via multiple attribute values, and the classical equivalence relation is replaced with the broader fuzzy proximity relation. We analyze in detail the process of attribute-oriented induction by concept hierarchies, utilizing the original properties of fuzzy databases to support this established data mining technique. In our approach we take full advantage of the implicit knowledge about the similarity of original attribute values, included by default in the investigated fuzzy database schemas. © 2007 Wiley Periodicals, Inc.

1. INTRODUCTION

Attribute-oriented induction (AOI) is a descriptive database mining technique, which gradually transforms the original, massive set of data (i.e., *initial relation*) into a concise form at the higher abstraction level, called a *generalized relation*. The induction is achieved via hierarchical aggregation of similar data collections, stored originally in a database at the low (primitive) level. The different attribute values are merged into the common and more abstract conceptual representations based on the background knowledge reflected in the concept hierarchy. The aggregation of database records in the AOI approach is performed on an attribute-by-attribute basis, where a separate concept hierarchy is employed for generalization of each of the attributes included in the relation of task-relevant data. Those hierarchies are indispensable for the AOI technique and were originally considered as a part of expert's knowledge about generalized attributes. In

*Author to whom all correspondence should be addressed: e-mail: angryk@cs.montana.edu.

†e-mail: fep@eecs.tulane.edu.

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 22, 763–779 (2007)
© 2007 Wiley Periodicals, Inc. Published online in Wiley InterScience
(www.interscience.wiley.com). • DOI 10.1002/int.20227



this article we investigate the character of the knowledge stored in the proximity relations of fuzzy databases and analyze possible ways of its application to the attribute-oriented generalization of the data originally stored in the form of fuzzy tuples.

Due to the immense magnitude of data currently stored in commercial and scientific databases, induction mechanisms became a practical necessity for many data analysts. To retrieve interesting patterns or association rules it is often indispensable to start from pruning and compressing the voluminous sets of the original data. Continuous processing of the original data is excessively time consuming and might be expendable, if we are actually interested only in information on abstraction level much higher than directly reflected by the technical details stored usually in large databases (e.g., serial numbers, time of transactions with precision in seconds, detailed GPS locations, etc.). Simultaneously, the data themselves conceptually represent information at multiple levels (e.g., a bar code, 040101000, represents a Milky Way Bar, which is a Chocolate Bar, pertaining to a Snack, or even a Food group, etc.). This makes the original data naturally appropriate for generalization allowing its transformation to a certain level of abstraction, dictated by the character of the analyzed data set or by preferences of data analysts and their clients.

Depending on the approach and the intention of data analysts, generation of the generalized relation can be treated either as a final step of data mining (i.e., the table is presented to the clients as a data summary, allowing them to interpret overall information¹⁻⁴) or as a base for further knowledge extraction (e.g., extraction of abstract association rules directly from the generalized relation⁵⁻⁷).

The hierarchical character of AOI provides analysts with the opportunity to view the original information at multiple levels of abstraction, allowing them to progressively discover interesting data aggregations. This approach seems to be much more appropriate for detailed analysis of data sets than commonly used simplified, nonhierarchical summarization. In contrast to the flat summarization, a gradual process of AOI through concept hierarchies allows detailed tracking of all records and can lead to the discovery of interesting patterns among data at the lowest abstraction level of their occurrence. As a result, we are able to avoid unnecessary loss of information due to overgeneralization. Moreover, in the AOI approach, the tuples, which did not achieved significant aggregation at a low abstraction level, rather than being removed from analysis, are gradually further aggregated and so given a chance to reach a meaningful count at one of higher abstraction levels.

In the next sections we introduce attribute-oriented induction and briefly characterize crisp and fuzzy approaches to the data generalization; we will also discuss the unique features of fuzzy database schemas that were utilized in our research on attribute-oriented generalization. In the third part we will present three techniques allowing convenient generalization of records stored in fuzzy databases. The increase in efficiency of these methods over the originally proposed solutions is achieved by taking full advantage of the knowledge about generalized domains stored implicitly in fuzzy database models. Then we will propose a method that allows multicontextual generalization of tuples in the analyzed database environments.

2. BACKGROUND

2.1. Attribute-Oriented Induction

The idea of generalization of the database records through extensive utilization of the concept hierarchies was popularized by Han and his coresearchers^{1-5,7-10} and extended further by Hamilton, Hilderman, and their coworkers.¹¹⁻¹² The majority of this work focuses on attribute-oriented induction with utilization of crisp concept hierarchies, where each attribute variable (concept) at each level of hierarchy can have only one direct abstract (its direct generalization) to which it fully belongs (there is no consideration of the degree of relationship, e.g., {master of art, master of science, doctorate} \subset graduate, {freshman, sophomore, junior, senior} \subset undergraduate).

Eight strategies applicable to the hierarchical generalization of attribute values have been proposed.³ The most essential for our analysis is the principle of vote propagation. It states that each tuple coming from the initial task-relevant relation should be represented by a single vote during the whole induction process. The votes should be accumulated when merging identical tuples at each step of AOI and this count, which reflects the number of the original tuples generalized to the particular abstract concept, should be stored in each abstract tuple and then reported as an additional argument in the generalized relation. Because the count of the votes allows preservation of the original dependencies occurring among the real (low-level) data at each stage of induction, its correctness is crucial for the exactness of AOI performed. The importance of this observation as well as the preservation of exact vote propagation when utilizing fuzzy concept hierarchies will be explained in the subsequent parts of this section.

Recently multiple groups of researchers investigated applications of fuzzy concept hierarchies for AOI. Lee and Kim¹³ used fuzzy ISA hierarchies, from the area of data modeling, to generalize database records to more abstract concepts. Lee¹⁴ applied fuzzy generalization hierarchies to mine generalized fuzzy quantitative association rules. Cubero et al.¹⁵ presented fuzzy gradual rules for data summarization. Raschia and Mouaddib¹⁶ implemented the SaintEtiq system for data summarization through extended concept hierarchies.

A fuzzy hierarchy of concepts reflects the degree with which a concept belongs to its direct abstract. In addition, more than one direct abstract of a single concept is allowed during fuzzy induction. Each link ℓ in the fuzzy concept hierarchy is a bottom-up directed arc (edge) between two nodes with a certain weight assigned to it. Such a structure reflects a fuzzy induction triple $(c^k, c^{k+1}, \mu_{c^k c^{k+1}})$, where c^k (i.e., an attribute value generalized to the k th abstraction level), and c^{k+1} (one of its direct generalizers) are endpoints of ℓ , and $\mu_{c^k c^{k+1}}$, being the weight of the link ℓ , represents the strength of conviction that the concept c^k (a source of ℓ) should be qualified as concept c^{k+1} (a target of ℓ) when the data generalization process moves to the next abstraction level. During attribute-oriented fuzzy induction (AOFI), the value of $\mu_{c^k c^{k+1}}$ dictates actually what fraction of a vote, representing original attribute values generalized already to the concept c^k , is to be propagated to its higher level representation, c^{k+1} .

Fuzzy hierarchies of concepts allow a more flexible representation of real-life dependencies. A significant weakness of this approach is a lack of automatic preservation of *exact vote propagation* at each stage of attribute-oriented fuzzy induction (AOFI). To guarantee exactness of the data summarization via AOFI we need to assure that each record from the original database relation will be counted *exactly once* at each of the levels of the fuzzy hierarchy.

This issue was successfully resolved by utilization of consistent fuzzy concept hierarchies,^{17,18} which preserve *exact vote propagation* due to their completeness and consistency. Speaking formally, to preserve completeness and consistency of the AOFI process for all adjacent levels, k and $k + 1$, in fuzzy concept hierarchy the following relationship must be satisfied:

$$\sum_{j=1}^{\|C^{k+1}\|} \mu_{c^k c_j^{k+1}} = 1.0, \quad c^k \in C^k, \quad c_j^{k+1} \in C^{k+1}$$

In other words, the sum of weights assigned to the links leaving any single node in a fuzzy concept hierarchy needs to be 1.0. Preservation of the above property prevents any attribute value (or its abstract) from being counted more or less during the process of AOFI (i.e., consistency of the fuzzy induction model is maintained). It also guarantees that a set of abstracts concepts at each level of hierarchy will cover all of the attribute values that occurred in the original data set (i.e., completeness of the model is maintained). In effect, we are guaranteed to not lose count of the original tuples when performing fuzzy induction on their attribute values.

Formally, each fuzzy concept hierarchy can be transformed to the consistent and complete generalization model via simple normalization of membership values in all outgoing links of the hierarchical induction model.

2.2. Similarity- and Proximity-Based Fuzzy Relational Databases

The similarity-based fuzzy model of a relational database, proposed originally in Refs. 19 and 20, is actually a formal generalization of the ordinary relational database.²¹ The fuzzy model, based on the max-min composition of a fuzzy similarity relation, which replaces the classical equivalence relation coming from the theory of crisp sets, was further extended by Shenoj and Melton,²² Shenoj et al.,²³ and De Kumar et al.²⁴ with the concept of the proximity relation. Because a fuzzy proximity relation has a more general character than a similarity relation, we will employ this model in our approach.

Important aspects of fuzzy relational databases are (1) allowing nonatomic domain values, when characterizing attributes of a single entity and (2) generation of equivalence classes based on the specific fuzzy relations applied in the place of traditional identity relation.

As mentioned above, each attribute value of the fuzzy database record is allowed to be a subset of the whole base set of attribute values describing a particular domain. Formally, if we denote a set of acceptable values for a single attribute as D , and we let d_{ij} symbolize a particular (j th) attribute value, characterizing the

i th entity, then instead of $d_{ij} \in D$, characteristic for the Codd's²¹ database model, the more general case $d_{ij} \subseteq D$ is allowed. That is, any member of the power set of accepted domain values can be used as an attribute value except the null set. A fuzzy database relation is a subset of the cross product of all power sets of its constituent attributes $2^{D_1} \times 2^{D_2} \times \dots \times 2^{D_m}$. This allows representation of inexactness arising from the original source of information. When a particular entity's attribute cannot be clearly characterized by a single descriptor, this uncertainty aspect can be reflected by multiple attribute values.

Another feature characterizing proximity fuzzy databases is substitution of the ordinary equivalence relation, defining the notion of redundancy in the ordinary database, with an explicitly declared proximity relation of which both the identity and similarity relations are actually special cases. Because the original definition of fuzzy proximity relations (also called tolerance relations) was only reflexive and symmetric, which is not sufficient to effectively replace the classical equivalence relation, the transitivity of proximity relation was added.²² This was achieved by extending the original definition of a fuzzy proximity relation to allow transitivity via similarity paths (sequences of similarities), using Tamura chains.²⁵ So the α -proximity relation used in proximity databases has the following structure.

If P is a proximity relation on a set of acceptable attribute values D , then given an $\alpha \in [0,1]$, two elements $x, z \in D$ are α -similar (denoted by $xP_\alpha z$) if and only if $P(x, z) \geq \alpha$ and are said to be α -proximate (denoted by $xP_\alpha^+ z$) if and only if they are (1) either α -similar or (2) there exists a sequence $y_1, y_2, \dots, y_m \in D$, such that $xP_\alpha y_1 P_\alpha y_2 P_\alpha \dots P_\alpha y_m P_\alpha z$.

Each of the attributes in the fuzzy database has its own *proximity table*, which includes the *degrees of proximity* (α -similarity) between all values occurring for the particular attribute. A proximity table for the domain of COUNTRIES, which we will use as an example for our further analysis, is presented in the Table I.

The proximity table can be transformed by Tamura chains to represent such an α -proximity relation. Results of this transformation are seen in Table II.

Now *the disjoint classes* of attribute values, considered to be equivalent at a specific α -level, can be extracted from the table. They are marked by shadings in the Table II. Such separation of the equivalence classes arises mainly due to the sequential similarity proposed by Tamura. For instance, despite the fact that the proximity degree, presented in Table I, between the concepts *Canada* and *Venezuela* is

Table I. Proximity table for a domain COUNTRY.

	Canada	USA	Mexico	Colombia	Venezuela	Australia	New Zealand
Canada	1.0	0.8	0.5	0.1	0.1	0.0	0.0
USA	0.8	1.0	0.8	0.3	0.2	0.0	0.0
Mexico	0.5	0.8	1.0	0.4	0.2	0.0	0.0
Colombia	0.1	0.3	0.4	1.0	0.8	0.0	0.0
Venezuela	0.1	0.2	0.2	0.8	1.0	0.0	0.0
Australia	0.0	0.0	0.0	0.0	0.0	1.0	0.8
New Zealand	0.0	0.0	0.0	0.0	0.0	0.8	1.0

Table II. α -Proximity table for a domain COUNTRY.

	Canada	USA	Mexico	Colombia	Venezuela	Australia	New Zealand
Canada	1.0	0.8	0.8	0.4	0.4	0.0	0.0
USA	0.8	1.0	0.8	0.4	0.4	0.0	0.0
Mexico	0.8	0.8	1.0	0.4	0.4	0.0	0.0
Colombia	0.4	0.4	0.4	1.0	0.8	0.0	0.0
Venezuela	0.4	0.4	0.4	0.8	1.0	0.0	0.0
Australia	0.0	0.0	0.0	0.0	0.0	1.0	0.8
New Zealand	0.0	0.0	0.0	0.0	0.0	0.8	1.0

0.1, the α -proximity is 0.4. Using the sequence of the original proximity degrees, $CanadaP_{\alpha}Mexico = 0.5 \wedge MexicoP_{\alpha}Colombia = 0.4 \wedge ColombiaP_{\alpha}Venezuela = 0.8$, we obtain $CanadaP_{\alpha}^{+}Venezuela = 0.4$, as presented in Table II.

The transformation based on the sequences of proximities converts the original proximity table back to a similarity relation as in a similarity database model.¹⁹ The practical advantage of the proximity approach comes from the lack of necessity to preserve the *max-min transitivity* when defining the proximity degrees. This makes a proximity table much easier for a user to define. The α -proximity table, although based on the proximity table, is generated dynamically only for the attribute values that were actually used in the fuzzy database.

3. ATTRIBUTE-ORIENTED GENERALIZATION IN FUZZY RELATIONAL DATABASES

Because a fuzzy database is an extension of the ordinary relational database model, the generalization of fuzzy tuples through the concept hierarchies can always be performed by the same procedure as presented by Han et al.³ for ordinary relational databases. However, here we focus on the utilization of the unique features of fuzzy databases in the induction process. First of all, due to the nature of fuzzy databases we have an ability to reorganize the original data to the required level of detail (by the merge of records, considered to be identical at a certain α -cut level, according to the given similarity or proximity relation), so then we can start attribute-oriented generalization from the desired level of detail. This approach, in removing unnecessary detail, must be applied with caution. When merging the tuples in fuzzy databases according to the equivalence at the given similarity level (e.g., by using SELECT queries with a high threshold level), we are not able to keep track of the number of original data records (itemsets) to be merged to a single tuple. Lack of such information may result in significant change of balance among the tuples in the database and lead to the erroneous (not reflecting reality) information presented later in the form of support and confidence of the extracted knowledge. This problem, which we call a *count dilemma*, is derived from the principle of vote propagation and can be easily avoided by performing extraction of initial data table at the very detailed level (i.e., $\alpha = 1.0$), where only identical

values are merged (e.g., values *England* and *Great Britain* will be unified). So then no considerable number of records would be lost as the result of such redundancy removal.

In this article we investigate three closely related approaches to AOFI, which are very convenient for implementation in the fuzzy database environments. The connection among these techniques derives from utilization of a proximity (or similarity) relation, a characteristic element of the fuzzy database models. We use it to support data generalization in these environments, emphasizing its use as a replacement for background knowledge, essential to generalization in other database models.

3.1. Extraction of Concept Hierarchies from α -Proximity Tables

The generation of an α -proximity relation for a particular domain D_j , in addition to providing a fuzzy alternative to an equivalence relation, also allows extraction of a partition tree, which can then be successfully employed by nonexperts to perform attribute-oriented induction.

From the propagation of shadings in the Table II, we can easily observe that the equivalence classes marked in the table have a nested character. As in Ref. 26, each α -cut (where $\alpha \in (0, 1]$) of a fuzzy binary relation in Table II creates disjoint equivalence classes in the domain D_j . If we let Π_α denote a single equivalence class partition induced on the domain D_j by a single α -level-set, through the increase of the value of α to α' we are able to extract the subclass of Π_α , denoted $\Pi'_{\alpha'}$ (a refinement of the previous equivalence class partition). A nested sequence of partitions $\Pi_{\alpha^1}, \Pi_{\alpha^2}, \dots, \Pi_{\alpha^k}$, where $\alpha^1 < \alpha^2 < \dots < \alpha^k$ may be represented in the form of a partition tree, as in Figure 1.

This nested sequence of partitions in the form of a tree has a structure identical with that of a crisp concept hierarchy applicable for AOI. Because the AOI approach is based on the reasonable assumption that an increase of degree of abstraction makes particular attribute values appear identical, the increase of conceptual abstraction in the concept hierarchy tree can be as well reflected by the decrease of α values (representing degrees of similarity between original attribute values). The lack of abstraction (0-abstraction level) at the bottom of

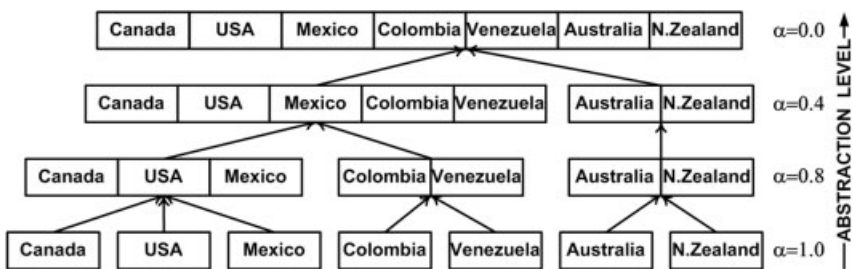


Figure 1. Partition tree of domain COUNTRY, built on the basis of Table II.

Table III. Table of abstract descriptors (for Figure 1).

Attribute value	Abstraction level (α)	Abstract descriptor
Canada	0.8	N. America
Colombia	0.8	S. America
Australia	0.8	Oceania
Canada	0.4	Americas
Australia	0.4	Oceania
Canada	0.0	Any

generalization hierarchy complies with the 1-cut of the α -proximity relation ($\alpha = 1.0$) from the fuzzy model and can be denoted as $S_{1.0}$. In other words, it is the level where only those attribute values that have identical meaning (i.e., synonyms) are aggregated.

The only thing differentiating the hierarchy in Figure 1 from the crisp concept hierarchies applicable for AOI is the lack of abstract concepts, which are used as the labels characterizing the sets of generalized (grouped) concepts. To create a complete set of the abstract labels, it is sufficient to choose *only one* member of every equivalence class Π (i.e., a single original value of the attribute) at each level of hierarchy (reflected by α), and assign a unique abstract descriptor to it. Sets of such definitions (original value of attribute and value of α linked with the new abstract name) can be stored as a database relation (Table III), where the first two attributes create a natural key for this relation.

The combination of partition tree in Figure 1 and the relation of abstract descriptors (Table III) allow us to create the classical generalization hierarchy in the form of Figure 2.

The disjoint character of equivalence classes generated from the α -proximity (i.e., similarity) table does not allow any concept in the hierarchy to have more than one direct abstract at every level of generalization hierarchy. Therefore this approach can be utilized only to form a crisp generalization hierarchy. Such a hierarchy, however, can then successfully be applied as a foundation to the development of a fuzzy

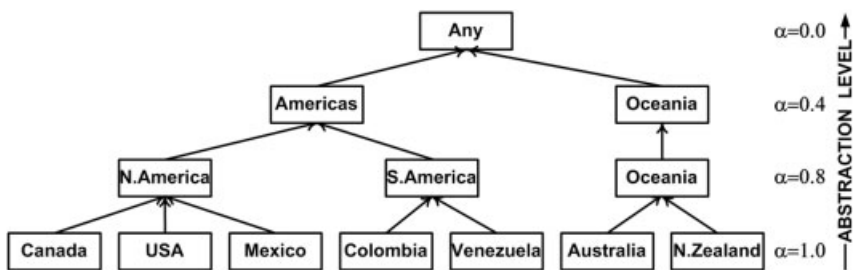


Figure 2. Crisp generalization hierarchy formed using Tables II and III.

concept hierarchy—by extending it with additional edges to represent partial membership of the lower level concepts in their direct abstract descriptors. Depending on the values of assigned memberships, data analysts can generate consistent or inconsistent fuzzy concept hierarchies.

3.2. Generation of Single-Level Hierarchies

Because both the similarity and the proximity relation are binary fuzzy relations, they can be interpreted in terms of fuzzy classes, where memberships of other elements (in our case other attribute values) in the fuzzy class $P(x)$ are derived from the rows in the proximity (or similarity) relation.²⁷ In other words, the grade of membership of attribute value y in the fuzzy class $P(x)$ (e.g., a fuzzy representative of attribute value x or of the set of such values), denoted by $\mu_{P(x)}(y)$, is $xP_{\alpha}y$ (or $xP_{\alpha}^{+}y$ for the similarity relation). For instance, from the Table I we can extract a definition of the fuzzy class *Australia*, that is, $P(\textit{Australia}) = \{\textit{Australia}|1.0, \textit{New Zealand}|0.8, \textit{Canada}|0.0, \textit{Mexico}|0.0, \textit{Colombia}|0.0, \dots\}$. This alternative view on fuzzy binary relations provides motivation for development of two other approaches. These may be more convenient for a user who does not have the expertise to carry the whole induction process from the basics and would like to use some support derived from existing proximity tables. Both of the techniques to be described require some knowledge about generalized domains, but not at an expert's level.

3.2.1. Typical-Element Based Generalization

In practice, when generalizing attribute values, one can usually distinguish subsets of lower level concepts that can be assigned to the particular abstract concepts easily even by users who have only ordinary knowledge about the generalized domain. At the same time the assignment of the other lower level values is problematic even among experts. Using the terminology presented in Ref. 28 we can say that each attribute has a domain (allowed values), a range (actually occurring values), and a typical range (most common values). We apply this approach to the generalization process. When we have an abstract concept we can often identify its *typical direct specializers*, which are the elements clearly belonging to it (e.g., we all would probably agree here that country *Holland* can be generalized to the concept *Europe* with 100% surety). This can be represented as a core of the fuzzy set (abstract concept) characterized by a membership of 1.0. However, there are usually also lower level concepts, which cannot be definitely assigned to only one of their direct abstracts (e.g., assigning the *Russian Federation* fully to the abstract concept *Asia* would probably raise some doubts, as almost one-fifth of this country lies west of the Urals). We call such cases *possible direct specializers* and define them as the concepts occurring in the group of lower level concepts characterized by the given abstract descriptor (fuzzy set) with the membership $0 < \mu \leq 1$. Such elements create the support of a fuzzy set and are to be interpreted as the range of the abstract concept.

In short, the idea behind this approach is to define initially the abstract concepts via choosing their basic representative attribute values (i.e., typical representative specializers) and then to use the proximity table to extract a more precise definition of the abstract class. For such extraction we assume a certain level of proximity (α), which should be interpreted as a level of precision reflected in our abstract concept definition.

Using this approach we initially characterize each new abstract concept as a set of its typical original attribute values with the level of doubt about its other possible specializers reflected by the value of α . Then we select the fuzzy class created from the α -cut of proximity relation for these predefined typical specializers and assess if this class fits well our expectations. Obviously some background knowledge about a generalized domain is now necessary so that the data analyst could point out typical direct specializers. However, there is still a significant support provided by the proximity table allowing smooth generalization of the most problematical cases. When we characterize an abstract concept by more than one typical element, we must also choose an intersection operator that best fits our preferences. This is necessary so that we would be able to extract a unified definition in the case when the typical elements (i.e., fuzzy classes) are overlapping.

For instance, if we predefine the abstract concept *North America* by its two typical countries *Canada* and *USA* with the level of proximity $\alpha = 0.6$, and with the operator *MAX* to reflect our optimistic approach, we can derive from Table I that

$$\begin{aligned} \text{North America} &= \text{MAX}(\text{Canada}, \text{USA})_{0.6} \\ &= \{\text{Canada}|1.0; \text{USA}|1.0; \text{Mexico}|0.8\} \end{aligned}$$

Defining now *Middle America* as the cut of fuzzy class *Colombia* on the 0.4 level we can extract from the proximity table:

$$\text{Middle America} = \text{Colombia}_{0.4} = \{\text{Mexico}|0.4; \text{Colombia}|1.0; \text{Venezuela}|0.8\}$$

As a result we can derive the fuzzy concept hierarchy and even modify the generalization model to become consistent through the normalization of derived memberships:

$$\begin{aligned} \text{North America} &= \left\{ \text{Canada}|1.0; \text{USA}|1.0; \text{Mexico} \left| \frac{0.8}{0.8 + 0.4} \right. \right\} \\ &= \{\text{Canada}|1.0; \text{USA}|1.0; \text{Mexico}|0.67\} \\ \text{Middle America} &= \left\{ \text{Mexico} \left| \frac{0.4}{0.8 + 0.4} \right.; \text{Colombia}|1.0; \text{Venezuela}|0.8 \right\} \\ &= \{\text{Mexico}|0.33; \text{Colombia}|1.0; \text{Venezuela}|0.8\} \end{aligned}$$

As distinct from the previous approach, here users are given much more freedom in defining abstracts. It is totally their decision as to which lower level concepts will be aggregated and to what abstraction level. They can modify two elements to control results of the generalization: (1) sets of typical elements and

Table IV. Characteristics of high-level concepts extracted from the proximity table.

Quantity of typical elements	Proximity level (α)	
	Low (close to 0)	High (close to 1)
Small	Spread widely	Precise
Large	Confusing (Error suspected)	Precise (Assured)

(2) level of acceptable proximity (α). Depending on the choice, the extracted definitions of abstract concepts may have different characteristics. In Table IV we summarize observations about the nature of definitions generated with this approach. The most precise definitions are going to be generated when users provide one typical element and allow only for a small increase of abstraction by imposing a proximity level close to 1.0. The same minimum of imprecision in generated definitions can be also accomplished with multiple values (i.e., typical elements) if they all have identical meaning. Despite the fact that multiple synonyms additionally confirm a user’s intent, they should not cause a spread of the meaning of the generated definition. Obviously, a choice of acceptable proximity ranges usually has a significant impact on the generated results. Definitions based on different typical elements may appear as (1) distinct when we perform the α -cut on a proximity table extracting only values that have high similarity to the typical elements, and as (2) almost identical when we move to the higher abstraction level by defining larger range of acceptable α ’s and allowing less similar concepts to be placed in the extracted definitions.

When extracting overlapping definitions of abstract concepts from a proximity table, a fuzzy concept hierarchy results and one must be extremely careful to keep this semantically meaningful. For instance, it makes no sense to predefine two or more general concepts at a level of abstraction so high that they are interpreted almost as identical.

Some basic guidelines are necessary when utilizing this approach.

(1) We need to assure that the intuitively assumed value of α extracts the cut (subset) of attribute values that corresponds closely to the definition of the abstract descriptor we desired. The strategy for choosing the most appropriate level of α -cut when extracting the abstract concept definitions comes from the principle of minimal generalization (minimal concept tree ascension strategy in Ref. 3), which translates to minimal proximity level decrease in our approach. Accepting the strategy of minimal abstraction increase at each successive step of AOI, we would recommend always choosing the definition extracted at the highest acceptable level of proximity, in other words, the definition provided by the cut of the proximity relation with the biggest possible value of α , which already embraces all pre-defined typical components of desired abstract descriptor (i.e., at the level where the typical components occur the first time in the common equivalence class). This approach allows us to prevent unnecessary overgeneralization.

(2) The problem of selecting appropriate representative elements without external knowledge about a particular attribute still remains; however, it can now be supported by the study of the values stored in the proximity (or similarity) table. Choosing typical values and then extracting a detailed definition from the proximity table will make AOFI more accessible for nonexperts. By using common knowledge they may be able to point out typical elements of a generalized concept while lacking expert knowledge necessary to characterize particular abstract in detail.

(3) Moreover, we should be aware that if the low-level concepts, predefined as typical components of the particular abstract descriptor, do not occur in the common equivalence class at any proximity level, then the contexts of the generalized descriptor and the proximity relation may be not in agreement and revision of the proximity table (or the abstract concepts) is necessary.

This approach allows us to place at the same level of concept hierarchy abstract concepts that were built with different levels of proximity (different α -values). As a result this achieves more effective (i.e., compressed) induction. However, allowing such a situation, especially when using a similarity table, we always have to remember that the abstract concepts derived from the similarity relation have a nested character. Placement of one abstract concept simultaneously with the other, being its actual refinement, does not make sense and is in contradiction with the relation represented by the similarity table.

The approach presented above allows us to form only two-level generalization hierarchies or to derive the generalized concepts at the first level of abstraction in the concept hierarchy. Each of the abstract concepts defined with this method is a generalization of the original attribute values, and therefore cannot be placed at the higher level of the concept hierarchy.

The inability to derive multilevel hierarchical structures does not stop this approach from being appropriate and actually very convenient for rapid data summarization or something we call selective attribute-oriented generalization. To quickly summarize the given data set, we may actually prefer to not perform gradual (hierarchical) generalization, but to replace it with a two-level hierarchy covering the whole domain of attribute values. Such an appropriately built “flat hierarchy” would represent the majority of dependencies between the original low-level concepts, which are to be generalized, by propagation of fractions of vote coming from each single attribute value instead of performing detailed hierarchical generalization.

In selective generalization, we generalize all attribute values from a specific point of view that is dictated by the nature of the data mining task. We generalize data in this specific context, omitting the records that do not fall into it. An example of such a generalization is presented in Figure 3. Using that fuzzy hierarchy of concepts we generate a generalized table with only those tuples that include countries from the regions of North and Middle America.

3.2.2. *Single Attribute Value Oriented Generalization*

As in the previous approach, this technique also leads to the generation of a two-level hierarchy. It is derived from the same approach of fuzzy class extraction

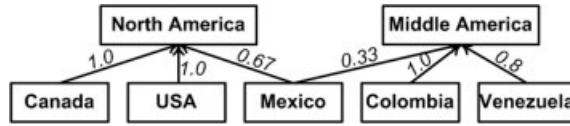


Figure 3. Consistent fuzzy generalization hierarchy built by extraction of partial knowledge stored in the proximity relation for the attribute COUNTRY.

from the similarity or proximity table as the technique above. The biggest advantage of this method is its ability to perform generalization of all original attribute values in the context represented by the proximity table, but from the point of view of the relevance of all these attribute values with respect to the distinguished one (i.e., the one being a foundation of the extracted fuzzy class).

Let us use a simple example to illustrate this. From the proximity relation (Table I) we can extract a single row, representing a fuzzy proximity class for the attribute value *Canada* (Table V).

Now we can generate subsets of this fuzzy class’s domain (which is actually the whole set of acceptable attribute values), defining disjoint ranges of acceptable membership values (i.e., proximity levels):

$$Canada\text{-similar} = \{Canada_{[0.6, 1.0]}\} = \{Canada, USA\}$$

$$Canada\text{-semisimilar} = \{Canada_{[0.1, 0.6]}\} = \{Mexico, Colombia, Venezuela\}$$

$$Canada\text{-unsimilar} = \{Canada_{[0.0, 0.1]}\} = \{Australia, New Zealand\}$$

Size of the assumed ranges is dependent on the preferences of the data analyst. Smaller ranges generate a larger number of abstract classes, but may reduce the number of lower level concepts in each of the extracted abstracts. The classes presented above give us sufficient information to perform generalization of all values occurring in the attribute COUNTRY from the point of view of the similarity of these values to the country *Canada* (Figure 4). Obviously values *Canada* and *USA* will be generalized as *Canada-similar* concepts, countries more distant as *semisimilar* and *unsimilar*. Because degrees of proximity were already utilized to extract these three subsets, we have not inserted them in the concept hierarchy.

Technically speaking, we simply sliced the single fuzzy class extracted from the proximity relation into layers reflecting levels of the relevance of the attribute

Table V. Fuzzy proximity class for the attribute value *Canada*.

	<i>Canada</i>	<i>USA</i>	<i>Mexico</i>	<i>Colombia</i>	<i>Venezuela</i>	<i>Australia</i>	<i>New Zealand</i>
<i>Canada</i>	1.0	0.8	0.5	0.1	0.1	0.0	0.0

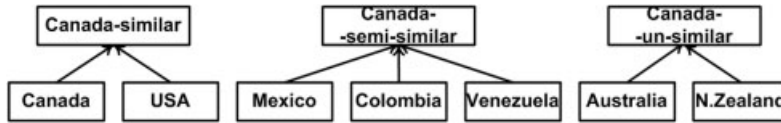


Figure 4. Value-similarity-based generalization.

values to the original attribute value, which then became a center of our attribute-oriented induction process.

3.3. Multicontext Generalization

The above approaches, despite being very convenient (and almost automatic) to use, have one common limitation. As discussed, both similarity and proximity relations can be successfully utilized when building concept hierarchies. However, the nature of the generalization based on these particular hierarchies can only reflect the context represented by the original similarity or proximity tables. Each of the attribute values can be considered basically as a point in a multidimensional space, reflecting multiple contexts of possible inductions (e.g., countries can be aggregated by province, and then by continent, or by the official native language, and then by the linguistic groups). Utilization of the predefined proximity tables restricts the context of an AOFI to the dimensionality originally reflected by these tables. This might be limiting for any users who want to perform the AOFI in a different context than represented in proximity tables. The techniques presented above do not provide flexibility in allowing such modifications. Obviously, the data-mining analysts can be allowed to modify the values in the proximity table in their own user views of the fuzzy database to represent the similarity between the concepts (attribute values) in the context of their interests. The advantage of such modifications is that now they can use these new proximity tables to merge the records that are identical in the context of their interests.

The task of dropping dimensions (contexts) of performed induction at some level of the generalization hierarchy, if not primarily reflected in the proximity relation, remains however an important issue. We describe here briefly an approach allowing a solution for this problem.

The last two methods presented in the previous section allowed only a single-step generalization. To extend AOI to a higher abstraction level we need to simply define new similarity or proximity tables reflecting similarity between the generalized concepts. There is nothing preventing users from defining such tables in the new context, as long as they have sufficient knowledge about a generalized domain to be able to correctly describe such dependencies. We can even halt the induction process performed with the multilevel hierarchy, presented in Figure 2, at any level and build new similarity (or proximity) tables reflecting the distance between abstract concepts in a totally different context than those primarily represented by the original hierarchy. In Table VI we present such a similarity relation, reflecting

Table VI. New similarity table reflecting the relation between concepts from the first abstraction level, according to the placement of Northern and Southern hemispheres of the Earth.

	<i>N. America</i>	<i>S. America</i>	<i>Oceania</i>
<i>N. America</i>	1.0	0.5	0.5
<i>S. America</i>	0.5	1.0	0.7
<i>Oceania</i>	0.5	0.7	1.0

similarity in context different than the one originally represented by the relation in Table II.

By cutting the hierarchy from Figure 2 after the first level of abstraction and introducing new abstract names, which better fit the relation presented in Table VI, we can generate a new concept hierarchy allowing induction from the first abstraction level to the third level in a new context. Then it will be merged with layers cut from Figure 2 to perform AOI based on the modification of generalization context at the first level of the abstraction. The hierarchy constructed in this manner is seen in Figure 5.

4. CONCLUSIONS

In this article have we presented three possible ways that similarity and proximity relations, implemented as the essential parts of fuzzy databases, can be successfully applied to data mining via attribute-oriented generalization. We demonstrated that both of these relations could be successfully utilized when building generalization hierarchies, assuming that both the context of the intended induction and the conceptual perspective reflected in the similarity or proximity table are in agreement. Moreover, we also considered how both of these fuzzy binary relations could be employed in building two-level concept hierarchies, allowing transformation of the original attribute values into a representative group of abstract concepts in a single step of the fuzzy induction. We characterized the

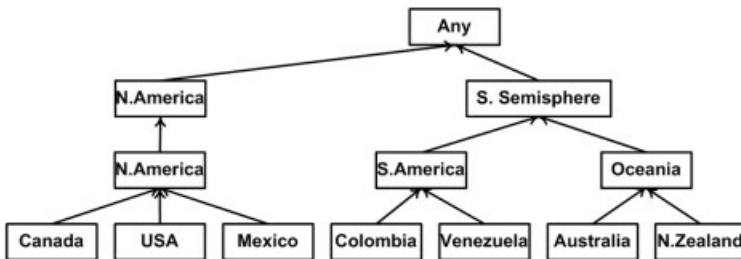


Figure 5. Multicontextual generalization hierarchy built from the merging of two similarity tables starting at different abstraction levels.

potential use of such approaches in rapid data summarization, selective generalization, or at the first stage of multicontextual generalization. We described the recursive manner in which the abstract concepts can be further generalized via building new proximity tables for the generalized attribute values, which may even reflect new contexts of induction, dependent on the knowledge supplied by the external source of expertise.

More advanced applications of the proposed solutions in the areas where fuzzy databases are the most applicable (i.e., spatial databases) remain for further work to illustrate a practical use of these approaches.

References

1. Cai Y, Cercone N, Han J. Attribute-oriented induction in relational databases. In: Proc IJCAI-89 Workshop on Knowledge Discovery in Databases, Detroit, MI; 1989. pp 26–36.
2. Han J, Fu Y. Exploration of the power of attribute-oriented induction in data mining. In: Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R, editors. Advances in knowledge discovery and data mining. Menlo Park, CA: AAAI/MIT Press; 1996. pp 399–421.
3. Han J, Cai Y, Cercone N. Knowledge discovery in databases: An attribute-oriented approach. In: Proc 18th Int Conf on Very Large Data Bases, Vancouver, Canada; 1992. pp 547–559.
4. Han J, Kamber M. Data mining: Concepts and techniques. New York: Morgan Kaufmann; 2000.
5. Han J. Mining knowledge at multiple concept levels. In: Proc 4th Int Conf on Information and Knowledge Management, Baltimore, MD; 1995. pp 19–24.
6. Srikant R, Agrawal R. Mining generalized association rules. In: Proc 21st Int Conf on Very Large Data Bases, Zurich, Switzerland; 1995. pp 407–419.
7. Han J, Fu Y. Discovery of multiple-level association rules from large databases. In: Proc 21st Int Conf on Very Large Data Bases, Zurich, Switzerland; 1995. pp 420–431.
8. Han J, Fu Y. Mining multiple-level association rules in large databases. *IEEE Trans Knowl Data Eng* 1999;11:798–804.
9. Han J, Nishio S, Kawano H, Wang W. Generalization-based data mining in object-oriented databases using an object cube model. *Data Knowl Eng* 1998;25:55–97.
10. Chen MS, Han J, Yu PS. Data mining: An overview from a database perspective. *IEEE Trans Knowl Data Eng* 1996;8:866–883.
11. Carter CL, Hamilton HJ. Efficient attribute-oriented generalization for knowledge discovery from large databases. *IEEE Trans Knowl Data Eng* 1998;10:193–208.
12. Hilderman RJ, Hamilton HJ, Cercone N. Data mining in large databases using domain generalization graphs. *J Intell Inform Syst* 1999;13:195–234.
13. Lee DH, Kim MH. Database summarization using fuzzy ISA hierarchies. *IEEE Trans Syst Man Cybern B* 1997;27:68–78.
14. Lee KM. Mining generalized fuzzy quantitative association rules with fuzzy generalization hierarchies. In: Proc Joint 9th IFSA World Congress and 20th NAFIPS Int Conf, Vancouver, Canada; 2001. pp 2977–2982.
15. Cubero JC, Medina JM, Pons O, Vila MA. Data summarization in relational databases through fuzzy dependencies. *Inform Sci* 1999;121:233–270.
16. Raschia G, Mouaddib N. SAINTETIQ: A fuzzy set-based approach to database summarization. *Fuzzy Set Syst* 2002;129:137–162.
17. Angryk RA, Petry FE. Consistent fuzzy concept hierarchies for attribute generalization. In: Proc IASTED Int Conf on Information and Knowledge Sharing, Scottsdale, AZ; 2003. pp 158–163.
18. Angryk RA, Petry FE. Data mining fuzzy databases using attribute-oriented generalization. In: Proc Third IEEE Int Conf on Data Mining, Workshop on Foundations and New Direction in Data Mining, Melbourne, FL; 2003. pp 8–15.

19. Buckles BP, Petry FE. A fuzzy representation of data for relational databases. *Fuzzy Set Syst* 1982;7:213–226.
20. Petry FE. *Fuzzy databases: Principles and applications*. Boston, MA: Kluwer Academic Publishers; 1996.
21. Codd FE. A relational model of data for large share data banks. *Commun ACM* 1970; 13:377–387.
22. Sheno S, Melton A. Proximity relations in the fuzzy relational database model. *Int J Fuzzy Set Syst* 1989;31:285–296.
23. Sheno S, Melton A, Fan LT. Functional dependencies and normal forms in the fuzzy relational database model. *Inform Sci* 1992;60:1–28.
24. De Kumar S, Biswas R, Roy AR. On extended fuzzy relational database model with proximity relations. *Fuzzy Set Syst* 2001;117:195–201.
25. Tamura S, Higuchi S, Tanaka K. Pattern classification based on fuzzy relations. *IEEE Trans Syst Man Cybern* 1971;SMC-1:61–66.
26. Zadeh LA. Similarity relations and fuzzy orderings. *Inform Sci* 1970;3:177–200.
27. Dubois D, Prade H. *Fuzzy sets and systems: Theory and applications*. New York: Academic Press; 1980.
28. Dubois D, Prade H, Rossazza JP. Vagueness, typicality and uncertainty in class hierarchies. *Int J Intell Syst* 1991;6:167–183.