University of Warsaw
Faculty of Mathematics, Informatics and Mechanics

mgr Andrzej Janusz

Algorithms for Similarity Relation Learning from High Dimensional Data

PhD dissertation

Supervisor

Prof. dr hab. Hung Son Nguyen

Institute of Mathematics
University of Warsaw

October 2013

Author's declaration:
aware of legal responsibility I hereby declare that I have written this dissertation myself and all the contents of the dissertation have been obtained by legal means.

October 31, 2013 .....

date *mgr Andrzej Janusz*

Supervisor's declaration:
the dissertation is ready to be reviewed

October 31, 2013 .....

date *Prof. dr hab. Hung Son Nguyen*

Abstract

The notion of similarity plays an important role in machine learning and artificial intelligence. It is widely used in tasks related to a supervised classification, clustering, an outlier detection and planning [7, 22, 57, 89, 153, 166]. Moreover, in domains such as information retrieval or case-based reasoning, the concept of similarity is essential as it is used at every phase of the reasoning cycle [1]. The similarity itself, however, is a very complex concept that slips out from formal definitions. A similarity of two objects can be different depending on a considered context. In many practical situations it is difficult even to evaluate the quality of similarity assessments without considering the task for which they were performed. Due to this fact the similarity should be learnt from data, specifically for the task at hand.

In this dissertation a similarity model, called Rule-Based Similarity, is described and an algorithm for constructing this model from available data is proposed. The model utilizes notions from the rough set theory [108, 110, 113, 114, 115] to derive a similarity function that allows to approximate the similarity relation in a given context. The construction of the model starts from the extraction of sets of higher-level features. Those features can be interpreted as important aspects of the similarity. Having defined such features it is possible to utilize the idea of Tversky's feature contrast model [159] in order to design an accurate and psychologically plausible similarity function for a given problem. Additionally, the dissertation shows two extensions of Rule-Based Similarity which are designed to efficiently deal with high dimensional data. They incorporate a broader array of similarity aspects into the model. In the first one it is done by constructing many heterogeneous sets of features from multiple decision reducts. To ensure their diversity, a randomized reduct computation heuristic is proposed. This approach is particularly well-suited for dealing with *the few-objects-many-attributes* problem, e.g. the analysis of DNA microarray data. A similar idea can be utilized in the text mining domain. The second of the proposed extensions serves this particular purpose. It uses a combination of a semantic indexing method and an information bireducts computation technique to represent texts by sets of meaningful concepts.

The similarity function of the proposed model can be used to perform an accurate classification of previously unseen objects in a case-based fashion or to facilitate clustering of textual documents into semantically homogeneous groups. Experiments, whose results are also presented in the dissertation, show that the proposed models can successfully compete with the state-of-the-art algorithms.

Keywords: Rule-Based Similarity, Similarity Learning, Rough Set Theory, Tversky's Similarity Model, Case-Based Reasoning, Feature Extraction

ACM Computing Classification (rev.2012): Computing methodologies ↦ Machine learning ↦ Machine learning approaches ↦ Instance-based learning.

Streszczenie

Pojęcie podobieństwa pełni istotną rolę w dziedzinach uczenia maszynowego i sztucznej inteligencji. Jest ono powszechnie wykorzystywane w zadaniach dotyczących nadzorowanej klasyfikacji, grupowania, wykrywania nietypowych obiektów oraz planowania [7, 22, 57, 89, 153, 166]. Ponadto w dziedzinach takich jak wyszukiwanie informacji (ang. information retrieval) lub wnioskowanie na podstawie przykładów (ang. case-based reasoning) pojęcie podobieństwa jest kluczowe ze względu na jego obecność na wszystkich etapach wyciągania wniosków [1]. Jednakże samo podobieństwo jest pojęciem niezwykle złożonym i wymyka się próbom ścisłego zdefiniowania. Stopień podobieństwa między dwoma obiektami może być różny w zależności od kontekstu w jakim się go rozpatruje. W praktyce trudno jest nawet ocenić jakość otrzymanych stopni podobieństwa bez odwołania się do zadania, któremu mają służyć. Z tego właśnie powodu modele oceniające podobieństwo powinny być wyuczane na podstawie danych, specjalnie na potrzeby realizacji konkretnego zadania.

W niniejszej rozprawie opisano model podobieństwa zwany Regułowym Modelem Podobieństwa (ang. Rule-Based Similarity) oraz zaproponowano algorytm tworzenia tego modelu na podstawie danych. Wykorzystuje on elementy teorii zbiorów przybliżonych [108, 110, 113, 114, 115] do konstruowania funkcji podobieństwa pozwalającej aproksymować podobieństwo w zadanym kontekście. Konstrukcja ta rozpoczyna się od wykrywania zbiorów wysokopoziomowych cech obiektów. Mogą być one interpretowane jako istotne aspekty podobieństwa. Mając zdefiniowane tego typu cechy możliwe jest wykorzystanie idei modelu kontrastu cech Tversky'ego [159] (ang. feature contrast model) do budowy precyzjnej oraz zgodnej z obserwacjami psychologów funkcji podobieństwa dla rozważanego problemu. Dodatkowo, niniejsza rozprawa zawiera opis dwóch rozszerzeń Regułowego Modelu Podobieństwa przystosowanych do działania na danych o bardzo wielu atrybutach. Starają się one włączyć do modelu szerszy zakres aspektów podobieństwa. W pierwszym z nich odbywa się to poprzez konstruowanie wielu zbiorów cech z reduktów decyzyjnych. Aby zapewnić ich zróżnicowanie, zaproponowano algorytm łączący heurystykę zachłanną z elementami losowymi. Podejście to jest szczególnie wskazane dla zadań związanych z problemem małej liczby obiektów i dużej liczby cech (ang. the few-objects-many-attributes problem), np. analizy danych mikromacierzowych. Podobny pomysł może być również wykorzystany w dziedzinie analizy tekstów. Realizowany jest on przez drugie z proponowanych rozszerzeń modelu. Łączy ono metodę semantycznego indeksowania z algorytmem obliczania bireduktów informacyjnych, aby reprezentować teksty dobrze zdefiniowanymi pojęciami.

Funkcja podobieństwa zaproponowanego modelu może być wykorzystana do klasyfikacji nowych obiektów oraz do łączenia dokumentów tekstowych w semantycznie spójne grupy. Eksperymenty, których wyniki opisano w rozprawie, dowodzą, że zaproponowane modele mogą skutecznie konkurować nawet z powszechnie uznanymi rozwiązaniami.

Contents

Chapter 1

Introduction

For many centuries the idea of similarity has inspired researchers from different fields, in particular philosophers, psychologists and mathematicians. Since Plato and his student, Aristotle, people have been trying to systematize the world around them by creating ontologies and grouping similar objects, living organisms or natural phenomena based on their characteristics. Over the years, many of the great discoveries have been made by scientists and inventors who noticed some resemblance between processes or objects, and on that basis formed a theory describing them.

Although human mind is capable of effortlessly assessing similarities between objects, there is no single methodology of selecting or building similarity models appropriate for a wide range of complex object classes and domains. This dissertation deals with a problem of learning a similarity relation or constructing a similarity function from data with a particular focus on high dimensional object domains. Apart from an overview of several well-known similarity learning methods, a rule-based model of similarity is proposed, whose flexibility allows to overcome many practical issues related with the commonly used approaches. This model and its two extensions, which are designed specifically to facilitate dealing with extremely high dimensional objects, are tested in extensive experiments in order to show their practical usefulness.

1.1 Motivation and Aims

The ability to identify similar objects is believed to play a fundamental role in the process of human decision making and learning [119, 125, 158]. Stefan Banach was known to say that:

> *"Good mathematicians see analogies. Great mathematicians see analogies between analogies."*

The notion of similarity itself, however, slips out from the formal scientific definitions [51, 159]. Despite this fact, similarity or reasoning by analogy is being utilized by numerous machine learning algorithms in applications ranging from a supervised classification to unsupervised clustering and an outlier detection [1, 93, 157]. Knowing how to discriminate similar cases (or objects) from those which are dissimilar in a desired context would enable a more accurate classification and detection of unusual or dangerous situations or behaviours. Unfortunately, due to difficulties related to

an a priori selection of a similarity model, which are particularly apparent when a metric space representation of objects is high dimensional, the performance of similarity-based machine learning algorithms may be limited [15].

A scope of this dissertation is a problem of learning how to recognize whether two objects are similar in a pre-specified context. A variety of methods have been used in order to construct similarity models and define a relation which would combine intuitive properties postulated by psychologists with a good performance in real-life applications. Among those a huge share was based on distance measures. In that approach, objects are treated as points in a metric space of their attributes and the similarity is a non-increasing function of the distance between them. Objects are regarded as similar if they are close enough in this space [15, 83, 160]. Such models may often be improved by assigning weights to attributes which express their importance to the model. Tuning those weights results in better fitting the relation to a data set and can be regarded as an example of similarity learning. Algorithms for a computationally efficient optimization of parameters for common similarity measures were investigated by numerous researchers, e.g. [21, 53, 89, 102, 149, 166, 170, 171].

One may argue that the relation of this kind is very intuitive because objects which have many similar values of attributes are likely to be similar. However, researchers like Amos Tversky [41, 83, 159] empirically showed that in some contexts, similarity does not necessarily have properties like symmetry or subadditivity which are implied by distance measures. This situation occurs particularly frequent when we compare objects of great complexity, often described by a large number of attributes. The explanation for this may lie in the fact that complex objects can be similar in some aspects and dissimilar in others. Hence, some additional knowledge about the context is needed to decide which of the similarity aspects are more important [42, 159].

Moreover, the dependencies between local and global similarities may be highly non-linear and in order to capture them it is necessary to extract some higher-level features of objects. Since there usually are numerous possible features to consider, this task can rarely be performed by human experts. Instead, the higher-level characteristics of objects and methods for their aggregation need to be derived from available data. Of course, as in all types of machine learning tasks, a similarity learning algorithm needs to balance complexity and efficiency [93, 157]. The construction of an overly complex similarity model will take too much time and resources to be applicable to real-life problems. Such a model may also be over-fitted to available data and yield poor performance in assessing the similarity of new objects.

The aim of this dissertation is to address those issues by proposing a similarity learning model called Rule-Based Similarity. The main motivation for that model comes from Tversky's works on the feature contrast model of similarity [159]. Instead of embedding objects into a metric space of their attributes, in the proposed approach the objects are represented by sets of higher-level features which can be more semantically meaningful than the low-level attribute values. In the model, such new features are defined by rules extracted from data, analogically to a rule-based object representation discussed in [128]. Unlike in that approach, however, in Rule-Based Similarity the new features are not treated as regular attributes but rather, they are regarded as arguments *for* or *against* the similarity of the compared objects. By combining the set representation with techniques developed within the theory of

rough sets, the model tries to aggregate those arguments and to express the similarity in a context dictated by a given task (e.g. supervised classification or semantic clustering), and by other objects present in the data. In this way, the resulting similarity function is more likely to reflect natural properties of similarity without loosing its practical usefulness and reliability.

Due to the subjectivity and complexity of the similarity notion, those appealing qualities can not be justified based only on theoretical properties and intuitions. The second goal of this dissertation is to provide results of thorough experiments in which the performance of Rule-Based Similarity was evaluated on many different data sets. Usefulness of this model in practical tasks, such as a supervised classification and an unsupervised cluster analysis, was compared with other similarity models as well as to the state-of-the-art in a given domain. The results of those tests may be used as arguments confirming the validity of the proposed model design.

1.2 Main Contributions

In the dissertation the problem of learning a similarity relation for a predefined data analysis task is discussed. Expectations regarding the construction and general properties of similarity models are formulated. Major challenges related to this problem are characterised and some practical solutions are proposed. Finally, the validity of the proposed methods is shown through extensive experiments on real-life data. Hence the main contributions of this dissertation are threefold:

1. A discussion on properties of the similarity relation from the point of view of data analysis and artificial intelligence.
2. A proposition of a similarity model and some construction algorithms that combine intuitive expectations with efficiency in practical applications.
3. An implementation and an experimental evaluation of the proposed similarity model on a wide range of data sets and in different use scenarios.

In particular, after reviewing observations of psychologists regarding the nature of the similarity, definition of a *proper similarity function* is proposed in Section 3.1.3. It aims at providing a more formal description of an abstract *similarity function* concept. Intuitively, pairs of objects for which a proper similarity function takes high values are more likely to be in the real similarity relation, relative to a predefined context. An example of such a context can be a classification of objects from the investigated domain. In that case, a similarity learning process can be guided by the fundamental properties of the similarity for classification, which are stated in Section 3.1.2.

The context of a similarity assessment is imposed by a purpose for which the evaluation is performed. It is also influenced by a presence of other objects. Those general observations together with the computational effectiveness constitute a basis for the desirable properties of similarity learning models which are given in Section 4.1. They are treated as requirements and a motivation for designing the similarity model which is the main scope of this dissertation.

The proposed Rule-Based Similarity (RBS) model and its two extensions are described in Section 4.3. Section 4.3.2 shows the construction of the basic version of RBS, designed for learning the similarity in a classification context from regular

data tables. Additionally, this section offers an intuitive interpretation of the model and explains its relations with the rough set theory. An important aspect of the construction of RBS is the computation of a decision reduct for each of the decision classes occurring in the data. This often needs to be done for data sets containing numerical attributes. Algorithm 2 shows how to compute a reduct in such a case. Some of the basic mathematical properties of the RBS similarity function are discussed in Section 4.3.3. In this section it is also shown that under certain conditions the proposed function is a proper similarity function for a similarity relation in the context of a classification.

The first extension of RBS, which is designed to efficiently handle extremely high dimensional data sets, is presented in Section 4.3.4. Its core is an algorithm for the computation of a diverse set of dynamic decision reducts (Algorithm 3). By combining randomization with the greedy heuristic for the computation of reducts this algorithm enables an efficient construction of robust sets of higher-level features. Due to the diversity of the sets, those features correspond to different similarity aspects. The similarity function which is proposed for this model, aggregates the local similarities analogically to aggregations of classifier ensembles.

The second of the proposed extensions is described in Section 4.3.5. The purpose of this model is to facilitate the similarity learning from textual corpora. Unlike the previous models, unsupervised RBS does not require information regarding decision classes and can be used for cluster analysis. To extract higher-level features it uses a combination of Explicit Semantic Analysis with a novel algorithm for the computation of information bireducts (Algorithm 4).

All the models proposed in this dissertation were thoroughly evaluated in experiments described in Chapter 5. RBS was compared to several other similarity learning techniques in the classification context on a variety of data tables. The tests were performed on benchmark tables (Section 5.1) as well as on real-life microarray data sets containing tens of thousands attributes (Section 5.2). Finally, tests with the unsupervised RBS were conducted and their results were described in Section 5.3.

Most of the partial results of this dissertation were presented at international conferences and workshops. They were published in conference proceedings and respectable journals. For example, the publications related to the construction and the applications of Rule-Based Similarity include [60, 61, 62, 64, 65, 67, 70]. There are also several other research directions of the author that had a significant influence on the design of the proposed similarity learning models. Among them, the most important considered the problem of feature selection and learning with ensembles of single and multi-label classifiers [63, 66, 68, 69, 71, 85, 141, 168]. Moreover, the research on unsupervised version of Rule-Based Similarity was largely influenced by the author's previous work on the semantic information retrieval and Explicit Semantic Analysis, which was conducted within the SYNAT project [72, 142, 155].

1.3 Plan of the Dissertation

The dissertation is divided into six chapters. This introductory chapter aims to provide a brief description of the considered problem and to help a reader with navigation through the remaining part of the text.

Chapter 2 is devoted to the theory of rough sets. Its main role is to introduce

the basic concepts and notations used in the subsequent chapters. It is divided into three main sections. Section 2.1 introduces the notions of information and decision systems (Subsection 2.1.1). It also discusses fundamental building blocks of the rough set theory such as the indiscernibility relation (Subsection 2.1.2) and the notions of a concept, decision logic language and rules (Subsection 2.1.3). Section 2.2 explains the rough set view on the approximation of vague or imprecise concepts. It gives the definition of a rough set and shows elementary properties of lower and upper approximations (Subsection 2.2.1). Then, in Subsections 2.2.2 and 2.2.3, there is a discussion on finding appropriate approximation spaces for constructing approximations of concepts and relations. The last section of the second chapter (Section 2.3) focuses on rough set methods for selecting informative sets of attributes. It gives definitions of the classical information and decision reducts in Subsection 2.3.1, and then it reviews several extensions of this important notion, such as approximate reducts, dynamic reducts (Subsection 2.3.2) and a novel concept of decision bireducts (Subsection 2.3.3).

Chapter 3 introduces similarity as a relation between objects and discusses its main properties. It also provides an overview of the most well-known similarity models and gives examples of their practical applications. The chapter is divided into three sections. The first one (Section 3.1) starts with a discussion on psychological properties of similarity as a semantic relation in Subsection 3.1.1. After this introduction, the importance of setting a similarity evaluation in a context which is appropriate for a task is highlighted in Subsection 3.1.2. This discussion is followed by definitions of a proper similarity function and similarity-based classification rules in Subsection 3.1.3 and then, an overview of similarity model evaluation methods is given in Subsection 3.1.4. The next section (Section 3.2) summarises the most commonly used similarity models. The distance metric-based similarity modelling is characterized in Subsection 3.2.1. Then, Subsection 3.2.2 explains Tversky's feature contrast model as an alternative to the distance-based approach. The section ends with a brief description of hierarchical similarity modelling methods in Subsection 3.2.3. The chapter concludes with Section 3.3, which is a survey on applications of similarity models in machine learning. It shows how the similarity can be employed for a predictive data analysis and visualization (Subsection 3.3.1) and briefly discusses the Case-Based Reasoning framework (Subsection 3.3.2). It ends with a usage example of similarity functions for unsupervised learning in cluster analysis (Subsection 3.3.3).

Chapter 4 focuses on similarity learning methods. Its first section (Section 4.1) defines the problem of similarity learning and lists some desirable properties of a good similarity learning model. Section 4.2 presents examples of four popular approaches to the similarity learning task. Subsection 4.2.1 summarises methods that use feature extraction techniques in order to improve a similarity model by selecting attributes which are relevant in a given context or by constructing new ones. Subsection 4.2.2 is an overview of a very popular approach that utilizes a genetic algorithm to tune parameters of a predefined similarity function. Then, Subsection 4.2.3 shows how a similarity relation can be induced and optimized in a tolerance approximation space. The last example, given in Subsection 4.2.4, concerns a specific task of using Explicit Semantic Analysis for learning a semantic representation of texts which can be used to better evaluate their similarity. Section 4.3 describes the idea of Rule-Based Similarity which is the main contribution of this dissertation. Subsection 4.3.1 discusses

intuitions and motivations for this model. The following subsections (Subsection 4.3.2 and 4.3.3) reveal construction details of the model and describe its properties. The next two subsections show how Rule-Based Similarity can be adjusted to efficiently learn the similarity in contexts defined by two different tasks related to analysis of high dimensional data. Namely, Subsection 4.3.4 focuses on similarity learning from high dimensional data for a classification purpose and Subsection 4.3.5 deals with the problem of unsupervised similarity learning for clustering of textual documents. The last subsection of the chapter (Subsection 4.3.6) summarises the proposed models.

Chapter 5 provides results of experiments in which the proposed model was tested on benchmark and real-life data sets. Each section of this chapter is devoted to a series of experiments on different types of data. Section 5.1 investigates the performance of Rule-Based Similarity in the context of classification on standard and high dimensional data tables. First, Subsection 5.1.1 describes the data sets used in this series of tests. Then, Subsection 5.1.2 briefly characterises the competing similarity models and Subsection 5.1.3 discusses the results of the comparisons between them. Section 5.2 presents the evaluation of the dynamic extension to Rule-Based Similarity on microarray data. This section starts with Subsection 5.2.1 which discusses general properties of microarrays as an example of extremely high dimensional data. Subsection 5.2.2 shows how efficient Dynamic Rule-Based Similarity can be for coping with the few-objects-many-attributes problem, in comparison to the state-of-the-art in the microarray data classification. The last section of this chapter (Section 5.3) presents an example of an application of the unsupervised extension to Rule-Based Similarity. At the beginning, Subsection 5.3.1 explains the methodology of the experiment and clarifies how the compared similarity models were evaluated. Subsection 5.3.2 characterizes the models which were used in the comparison and Subsection 5.3.3 summarises the results.

Finally, the last Chapter 6 concludes the dissertation. Section 6.1 draws a summary of the discussed problems and Section 6.2 proposes some directions for future development of the rule-based models of similarity.

1.4 Acknowledgements

I wish to thank my supervisor, Prof. Hung Son Nguyen, for his guidance and invaluable support in writing this dissertation. I am thankful to Prof. Andrzej Skowron for countless inspiring conversations and remarks. They have always been a motivation for my research and writing. I am very grateful to Dominik Ślęzak whose enthusiasm and excellent ideas have taught me to stay open-minded. I would also like to sincerely thank others from Institute of Mathematics, who I consider not only my colleagues but my friends. I have learnt a lot from their knowledge and experience.

My deepest gratitude, however, goes to my family, especially to my parents and two sisters who have supported me through my entire life. Non of the things that I achieved would be possible without them. Thank You!

The research was supported by the grants N N516 368334, N N516 077837, 2011/01/B/ST6/03867 and 2012/05/B/ST6/03215 from the Ministry of Science and Higher Education of the Republic of Poland, and the National Centre for Research and Development (NCBiR) under the grant SP/I/1/77065/10 by the strategic scientific research and experimental development program: "Interdisciplinary System for Interactive Scientific and Scientific-Technical Information".

Chapter 2

Theory of Rough Sets

The theory of rough sets, proposed by Zdzisław Pawlak in 1981 [108], provides a mathematical formalism for reasoning about imperfect data and knowledge [113, 114, 115]. Since their introduction, rough sets have been widely used in numerous real-life applications related to intelligent knowledge discovery, such as classification, clustering, approximation of concepts, discovering of patterns and dependencies in data [10, 71, 95, 112, 113, 114, 118, 130]. They were also used for hierarchical modelling of complex objects [10, 103], as well as approximation of relations and functions [112, 132, 156].

The notion of similarity has always been important for researchers in the field of rough sets. Several extensions of the classical discernibility-based rough sets were proposed, in which a similarity relation was used to generalized rough approximations [43, 45, 114, 120, 133, 145, 146]. Similarity was also utilized in order to explain relations between rough sets and fuzzy sets and interpret fuzziness in the rough set setting [172]. On the other hand, some similarity measures were motivated by the rough set theory [57].

In this dissertation similarity is viewed as a relation whose properties may vary depending on a specific context. Since without any additional knowledge the similarity can be regarded as an arbitrary relation, it needs to be learnt from available data. The similarity relation is vague in nature [42, 90, 159]. For this reason the rough set theory seems suitable for this purpose. It does not only offer intuitive foundations for modelling complex relations, but also provides practical tools for extracting meaningful features and defining important aspects of similarity between considered objects [82, 118]. Those aspects often correspond to higher-level characteristics or concepts which can also be vague. To better cope with such a multi-level vagueness there were proposed models that combine the rough set and fuzzy set theories into rough-fuzzy or fuzzy-rough models [33, 104, 105].

The similarity learning model described in this dissertation (Section 4.3) derives from the theory of rough sets. To better explain their construction, the following sections briefly overview selected aspects of the rough sets and introduce some basic notation used in the remaining parts of this thesis. Section 2.1 gives definitions of fundamental concepts, such as an *information system* or a *decision rule*. Section 2.2 provides an insight on approximation spaces and explains the basic principles of a rough set approximation. Section 2.3 describes a rough set approach to the problem

of data dimensionality reduction. In its last Subsection 2.3.3 the notion of reducts is extended to bireducts and some interesting properties of decision bireducts are discussed.

2.1 Introduction to Rough Sets

The theory of rough sets deals with problems related to reasoning about vagueness in data [108]. Its main assumption is that with every object of the considered universe Ω there is some associated information which can be represented in a tabular form as attribute-value entries. Available objects which are characterized by the same information are indiscernible - it is not possible to make any distinction between them. Those elementary sets of indiscernible objects are used to model uncertainty of vague concepts.

In this dissertation, every concept is associated with a set of objects $X \subset \Omega$. It is usually assumed that information regarding belongingness of objects to X is available for at least a finite subset of objects $U \subset \Omega$. This subset is called a *training set*. When solving practical problems we are often interested in finding an accurate but comprehensible description of a concept X in terms of features of objects from the training set U. Ideally, this description should fit to all objects from Ω. In the rough set terminology, the process of finding an appropriate description of a concept is referred to as an approximation of X. In a more general context of machine learning, this task is often called a *classification problem*. The most part of this dissertation is focusing on similarity models which can be used to facilitate the classification.

Within the rough set approach, vagueness or vague concepts correspond to sets of objects which can not be precisely described using available information. To enable reasoning about such concepts, they are associated with two crisp sets which can be unambiguously defined [113, 114, 115]. The first set is the largest possible subset of available data that contains only objects which surely belong to the concept. The second set is the smallest possible set which surely contains all objects belonging to the concept in the available data. Together, those two set allow to handle vagueness without a need for introducing artificial functions, as it is done in the fuzzy set theory [173]. This section overviews the basic notions of the rough set theory which are used in further parts of this dissertation.

2.1.1 Information and decision systems

In the rough set theory, available knowledge about object $u \in U$ is represented as a vector of information about values of its *attributes*. An attribute can be treated as a function $a : U \to V_a$ that assigns values from a set V_a to objects from U. In a vast majority of cases, such functions are not explicitly given. However, we can still assume their existence if for any object from U we are able to measure, compute or obtain in other way the corresponding values of its attributes.

All available information about objects from U can be stored in a structure called an *information system*. Formally, an information system $\mathbb{S}$ can be defined as a tuple:

$$\mathbb{S} = \big(U, A\big) \tag{2.1}$$

Table 2.1: An exemplary information system $\mathbb{S}$ (Table (a)) and a decision system $\mathbb{S}_d$ with a binary decision attribute (Table (b)).

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
u_1	1	2	2	0	0	1	0	1
u_2	0	1	1	1	1	0	1	0
u_3	1	2	0	1	0	2	1	0
u_4	0	1	0	0	1	0	0	1
u_5	2	0	1	0	2	1	0	0
u_6	1	0	2	0	2	0	0	2
u_7	0	1	1	2	0	2	1	0
u_8	0	0	0	2	1	1	1	1
u_9	2	1	0	0	1	1	0	0

(a)

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	d
u_1	1	2	2	0	0	1	0	1	1
u_2	0	1	1	1	1	0	1	0	1
u_3	1	2	0	1	0	2	1	0	1
u_4	0	1	0	0	1	0	0	1	0
u_5	2	0	1	0	2	1	0	0	1
u_6	1	0	2	0	2	0	0	2	0
u_7	0	1	1	2	0	2	1	0	1
u_8	0	0	0	2	1	1	1	1	0
u_9	2	1	0	0	1	1	0	0	0

(b)

where U is a finite non-empty set of objects and A is a finite non-empty set of attributes. The most common representation of the information system is a table whose rows correspond to objects from U and columns are associated with attributes from A. There are however some other information system representation forms [154]. A simple example of an information system represented in the tabular form is given in Table 2.1.a (on the left).

It is usually assumed that information about values of all the attributes from A can be obtained for any object, including those which are not present in U. In such a case, those attributes are often called *conditional attributes*. However, there might also exist some special characteristic of objects from U, which can be used to define a partitioning of U into disjoint sets. Such a characteristic may correspond to, e.g. belongingness of the objects to some concept. In this case, it is possible to define an attribute, called a *decision* or *class attribute*, that reflects this characteristic. In order to deliberately emphasize its presence, an information system with a defined decision attribute is called a *decision system* and is denoted by $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$, where $A \cap \{d\} = \emptyset$. A tabular representation of a decision system is sometimes called a *decision table* and the disjoint sets of objects with different values of the decision attribute are called *categories* or *decision classes*. Table 2.1.b shows an exemplary decision system $\mathbb{S}_d$ with a binary decision attribute d (on the right).

Unlike in the case of conditional attributes, a value of a decision attribute may be unknown for objects from $\Omega \setminus U$. Therefore, the approximation of concepts (a classification problem) can sometimes be restated as a prediction of decision attribute values for objects which are not included in the training set. In many practical applications, such as the topical classification of textual documents [68, 85], it might be convenient to define more than one decision attribute. In such a case, a decision system will be denoted by $\mathbb{S}_D = (U, A \cup D)$, where D is a set of decision attributes and $A \cap D = \emptyset$, and the prediction of the decision values will be called a *multi-label classification* problem.

In many practical applications the assumption regarding availability of information concerning values of conditional attributes in decision systems is not true.

Real-life decision systems often have *missing* attribute values and some dedicated techniques for analysing this kind of data have been developed within the theory of rough sets [46, 86, 152]. The reasons for lack of partial information about particular objects might be diverse. The semantics of different kinds of missing values have also been studied [48, 49, 86]. Although this problem remains a vital research direction, handling data with missing or vague information lies outside the scope of this dissertation.

2.1.2 Indiscernibility relation

In the rough set theory objects from U are seen through the information that can be used to describe them. This fact implies that in a case when information available for two different objects does not differ (i.e. values on all attributes are the same), those objects are regarded *indiscernible*.

Definition 2.1 (Indiscernibility relation)**.**
Let $\mathbb{S} = (U, A)$ be an information system and let $B \subseteq A$. We will say that $u_1, u_2 \in U$ are satisfying the indiscernibility relation IND_B with regard to the attribute set B iff they have equal attribute values for every $a \in B$:

$$(u_1, u_2) \in IND_B \Leftrightarrow \forall_{a \in B} a(u_1) = a(u_2).$$

Otherwise u_1 and u_2 will be regarded discernible.

It is easy to observe that the indiscernibility is in fact an equivalence relation in U (i.e. it is reflexive, symmetric and transitive). An indiscernibility class of an object u with regard to an attribute set B will be denoted by $[u]_B$:

$$[u]_B = \{u' \in U : \forall_{a \in B} \ a(u') = a(u)\} \, . \tag{2.2}$$

Therefore, using the indiscernibility relation it is possible to define a *granulation* of objects described by an information system $\mathbb{S}$ into disjoint subsets. For any $B \subseteq A$ it will be denoted by $U/B = \{[u]_B : u \in U\}$. For example, the indiscernibility class of an object u_1 with regard to $\{a_1, a_3\}$ in the information system from Table 2.1.a (on the left) is $[u_1]_{\{a_1,a_3\}} = \{u_1, u_6\}$ and $U/\{a_1, a_3\} = \big\{\{u_1, u_6\}, \{u_2, u_7\}, \{u_3\}, \{u_4, u_8\}, \{u_5\}, \{u_9\}\big\}$.

Many different equivalence relations in U can be defined using different attribute subsets. The indiscernibility relations with regard to single attributes can serve as a basis for the construction of equivalence relations defined by any subset of attributes. For any two subsets of attributes $B, B' \subseteq A$ and any $u \in U$, the following equations hold:

$$[u]_B = \bigcap_{a \in B} [u]_{\{a\}} \, , \tag{2.3}$$

$$[u]_{B \cup B'} = [u]_B \cap [u]_{B'} \, , \tag{2.4}$$

$$B \subseteq B' \Rightarrow [u]_{B'} \subseteq [u]_B \, . \tag{2.5}$$

When constructing an approximation of a concept it is important to investigate a relation between indiscernibility classes with regard to conditional attributes and with regard to decision attributes.

Definition 2.2 (Consistent decision system)**.**
A decision system $\mathbb{S}_d = (U, A \cup D)$ *will be called consistent iff*

$$\forall_{u \in U} [u]_A \subseteq [u]_D. \tag{2.6}$$

Otherwise $\mathbb{S}_d$ *will be called inconsistent.*

Several extensions of the indiscernibility notion can be found in the rough set literature. For example, generalizations based on a tolerance relation [133, 135] or a predefined similarity relation [45, 145, 146] have been proposed in order to define better approximations of concepts. In other approaches the definition of indiscernibility has been modified to facilitate generation of decision rules from incomplete data [49, 86].

2.1.3 Descriptions and rules

The rough set theory is often utilized to provide description of *concepts* from the considered universe. Any concept can generally be associated with a subset of objects from U which belong or match to it. In general, decision attributes in a decision system can usually be interpreted as expressing the property of belongingness to some concept. Given some information (e.g. in the form of a decision system) about characteristics (values of attributes) of objects corresponding to the considered concept one may try to describe it using a *decision logic language* [109].

Decision logic language L_A is defined over an alphabet consisting of a set of attribute constants (i.e. names of attributes from A) and a set of attribute value constants (i.e. symbols representing possible attribute values). The attribute and attribute value constants can be connected using the equity symbol $=$ to form attribute-value pairs ($a = v$, where $a \in A$ and $v \in V_a$), which are regarded as atomic formulas of the language L_A. The atomic formulas can be combined into compound formulas of L_A using connectives from a set $\{\neg, \wedge, \vee, \rightarrow, \equiv\}$ called negation, conjunction, alternative, implication and equivalence, respectively. If ϕ and ψ are in L_A, then $\neg(\phi)$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$ and $(\phi \equiv \psi)$ are in L_A. The atomic formulas of a compound formula (the attribute-value pairs) are often called *descriptors* and the formula itself is sometimes called a *description* of some concept.

The satisfiability of a formula ϕ from L_A by an object from an information system $\mathbb{S} = (U, A)$, which is denoted by $u \vDash_{\mathbb{S}} \phi$ or by $u \vDash \phi$ if $\mathbb{S}$ is understood, can be defined recursively:

1. $u \vDash (a = v) \Leftrightarrow a(u) = v$.
2. $u \vDash \neg\phi \Leftrightarrow$ not $u \vDash \phi$.
3. $u \vDash (\phi \wedge \psi) \Leftrightarrow u \vDash \phi$ and $u \vDash \psi$.
4. $u \vDash (\phi \vee \psi) \Leftrightarrow u \vDash \phi$ or $u \vDash \psi$.
5. $u \vDash (\phi \rightarrow \psi) \Leftrightarrow u \vDash (\neg\phi \vee \psi)$.
6. $u \vDash (\phi \equiv \psi) \Leftrightarrow u \vDash (\phi \rightarrow \psi)$ and $u \vDash (\psi \rightarrow \phi)$.

Each description (a formula) ϕ in a decision logic language L_A can be associated with a set of objects from U that satisfy it. This set is called a *meaning* of the formula in an information system $\mathbb{S} = (U, A)$ and is denoted by $\phi(U) = \{u \in U : u \vDash \phi\}$. Moreover, we will say that a formula ϕ is *true* or *consistent* in $\mathbb{S}$ if and only if its meaning is equal to the whole set U (i.e. $\phi(U) = U$). Otherwise a formula is *inconsistent* in $\mathbb{S}$.

It is worth noticing that an indiscernibility class of any object u described in $\mathbb{S} = (U, A)$ can be expressed as a meaning of a formula in the language L_A as $[u]_A = \phi(U)$, where $\phi = \big(a_1 = a_1(u) \wedge \ldots \wedge a_i = a_i(u) \wedge \ldots \wedge a_m = a_m(u)\big)$, and $m = |A|$. Based on equations 2.3, 2.4 and 2.5 this can be generalized to indiscernibility classes with regard to any subset of attributes. For example, in the information system $\mathbb{S}$ from Table 2.1.a the meaning of $\phi = (a_1 = 1 \wedge a_3 = 2)$ is $\phi(U) = \{u_1, u_6\} = [u_1]_{\{a_1,a_3\}}$. One example of a formula that is consistent in $\mathbb{S}$ is $(a_7 = 0 \vee a_7 = 1)$.

In the rough set data analysis, knowledge about dependencies between conditional attributes and decision attributes of a decision system are often represented using special formulas called *decision rules*.

Definition 2.3 (Decision rules)**.**
Let A and D be conditional and decision attribute sets of some decision system. Moreover, let $L_{A\cup D}$ be a decision logic language and π be a formula of $L_{A\cup D}$. We will say that π is a decision rule iff the following conditions are met:

1. $\pi = (\phi \to \psi)$,
2. *ϕ and ψ are conjunctions of descriptors,*
3. *ϕ is a formula of L_A and ψ is a formula of L_D.*

The right hand side of a decision rule $\pi = (\phi \to \psi)$ (i.e. ψ) will be called a *consequent* or a *successor* of a rule and the left hand side will be called an *antecedent* or a *predecessor* (i.e. ϕ). The antecedent of π will be denoted by $lh(\pi)$ and the consequent of π will be marked by $rh(\pi)$. It is important to note that the above definition of a decision rule is more specific than the original definition from [109]. In fact the definition used in this dissertation corresponds to *P-basic decision rules* from Pawlak's original paper.

Decision rules aim at providing partial descriptions of concepts indicated by the decision attributes. They can be learnt from a decision system and then used to predict decision classes of new objects, provided that values of conditional attributes of those objects are known. For example, from the decision system $\mathbb{S}_d$ shown in Table 2.1.b we can induce decision rules:

$$\pi_1 = \big((a_4 = 0 \wedge a_6 = 1) \to (d = 1)\big)$$

and

$$\pi_2 = \big((a_2 = 1 \wedge a_3 = 1) \to (d = 1)\big).$$

The meaning of π_1 in $\mathbb{S}_d$ is the set $\pi_1(U) = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\} = U \setminus \{u_9\}$, whereas the meaning of π_2 in $\mathbb{S}_d$ is $\pi_2(U) = U$. The first rule is inconsistent in $\mathbb{S}_d$, whereas the second rule is true in $\mathbb{S}_d$. However, the second rule is more general than the first one, since meanings of the antecedents of those rules have different

cardinalities: $|lh(\pi_1)(U)| = |\{u_1, u_5, u_9\}| = 3$, and $|lh(\pi_2)(U)| = |\{u_2, u_7\}| = 2$. We may say that those rules are true with different degrees in $\mathbb{S}_d$, thus their predictive power is different.

There is also a different type of rules within the rough set theory, which can be particularly useful for analysing dependencies in data with multiple decision values, namely, *inhibitory rules* [26].

Definition 2.4 (Inhibitory rules)**.**
Let A and D be conditional and decision attribute sets of a decision system. Moreover, let $L_{A\cup D}$ be a decision logic language and π be a formula of $L_{A\cup D}$. We will say that π is an inhibitory rule iff the following conditions are met:

1. $\pi = (\phi \to \neg\psi)$,
2. *ϕ and ψ are conjunctions of descriptors,*
3. *ϕ is a formula of L_A and ψ is a formula of L_D.*

An inhibitory rule tell us that an object which satisfies the predecessor of this rule[1] cannot belong to a pointed decision class. The inhibitory rules can be seen as a complement to decision rules as they often provide means to classify objects which are difficult to cover by the traditional rules [26]. They are particularly useful for constructing classifiers in a presence of a highly imbalanced distribution of decision values. It needs to be noted, however, that a cardinality of a set of all possible inhibitory rules for a given data is usually much greater than that of all decision rules.

Usefulness of a rule for prediction of decision classes of new objects (or just *classification*, in short) can be quantitatively assessed using rule quality measures. There exist many measures that aim at evaluating the strength of dependency between the antecedent and the consequent of rules [2, 24, 117]. However, the bigger part of them is based on the notions of rule's *support* and *confidence*. The support of a rule π is defined as:

$$supp(\pi) = \frac{|lh(\pi)(U)|}{|U|}$$

and the confidence of π is:

$$conf(\pi) = \frac{|lh(\pi)(U) \cap rh(\pi)(U)|}{|lh(\pi)(U)|} = 1 - \frac{|U \setminus \pi(U)|}{|lh(\pi)(U)|}\,.$$

From the second equation it follows that the confidence factor of a rule π equals 1 iff the rule is consistent in $\mathbb{S}_d$. To prove it, it is sufficient to show that $U \setminus \pi(U) = lh(\pi)(U) \setminus rh(\pi)(U)$. This equity, however, is a straight consequence of a definition of the meaning of an implication:

$$\begin{aligned} u \in \pi(U) \Leftrightarrow u \vDash \big(lh(\pi) \to rh(\pi)\big) \quad &\Leftrightarrow \quad u \vDash \big(\neg lh(\pi) \vee rh(\pi)\big) \\ &\Leftrightarrow \quad \big(u \in U \setminus lh(\pi)(U)\big) \vee \big(u \in rh(\pi)(U)\big). \end{aligned}$$

[1] In the remaining parts of this dissertation such objects will also be regarded to as *matching* the rule.

If so, then:

$$u \in \big(U \setminus \pi(U)\big) \;\Leftrightarrow\; u \in \Big(U \setminus \big(U \setminus lh(\pi)(U)\big)\Big) \cap \Big(U \setminus rh(\pi)(U)\Big)$$
$$\Leftrightarrow\; u \in \Big(lh(\pi)(U) \setminus rh(\pi)(U)\Big).$$

The confidence of a rule is often interpreted as an indicator whether the rule is true. We may say that a rule is true in a degree corresponding to its confidence. An example of a rule quality measure that, in a sense, combines the desirable properties of the support and confidence coefficients is *Laplace m-estimate* defined as $laplace_m(\pi) = \frac{|lh(\pi)(U) \cap rh(\pi)(U)| + m \cdot p}{(|lh(\pi)(U)| + m}$, where m and p are positive parameters. Values of m and p usually correspond to a number of decision classes, and prior probability of the $rh(\pi)$, respectively [34]. Unlike the confidence, this measure favours rules with a higher support.

Intuitively, the support of a rule expresses how large data fragment the rule describes, i.e. measures its generality, whereas the confidence says how often the rule truly indicates consequent for objects belonging to the meaning of its antecedent. For instance, the support of the rule π_1 from the previous example is $3/9 = 1/3$ and its confidence is $2/3$. At the same time the support and the confidence of π_2 are $2/9$ and 1, respectively. In order to compare those rules we may also use the Laplace m-estimate for $m = 2$ and $p = 0.5$: $laplace_2(\pi_1) = 3/5$ whereas $laplace_2(\pi_2) = 3/4$. Rough set methods usually derive rules using descriptions of indiscernibility classes in $\mathbb{S}_d$.

Each formula in the language L_A corresponds to a unique set of objects but there is no guarantee that for a given subset of objects $X \subset U$ there exists a formula ϕ whose meaning equals X. Moreover, several different formulas may have exactly the same meaning in $\mathbb{S}$. A set of objects represented in an information system $\mathbb{S}$ that can be exactly described by some formula in a language L_A is called a *definable set* in $\mathbb{S}$. More formally, the set X will be called definable in $\mathbb{S} = (U, A)$ iff there exists a formula ϕ of the language L_A, such that $\phi(U) = X$. Subsets of U that are not definable will be called *undefinable*. The family of all definable sets in $\mathbb{S}$ will be denoted by $DEF(\mathbb{S})$.

Concepts corresponding to undefinable sets can be approximated using definable sets. A typical task in the rough set data analysis is to find an optimal approximation of a predefined concept using knowledge represented by a decision system and describe it using formulas, such as decision and inhibitory rules. Such an approximation is usually expected to be accurate not only for known objects from U, but also for the new ones which were not available when the approximation was learnt. For this purpose many rough set techniques employ the Minimal Description Length (MDL) principle and constrain the language used to describe and reason about the data. This approach to the problem of approximating the undefinable sets is the most characteristic feature of the rough set theory [110, 115].

2.2 Rough Set Approximations

In the rough set theory any arbitrary set of objects X can be approximated within an information system $\mathbb{S} = (U, A)$ by a pair of definable sets $App(X) = (\underline{X}, \overline{X})$, called a

rough set of X in $\mathbb{S}$. The set $\underline{X}$ is the largest definable set which is contained in X. Analogically, the set $\overline{X}$ is the smallest definable set which contains X. The sets $\underline{X}$ and $\overline{X}$ are called a *lower* and *upper approximation* of X in $\mathbb{S}$, respectively.

2.2.1 Lower and upper approximations

The lower and upper approximations can also be constructively defined using the notion of indiscernibility classes. Let $X \subseteq \Omega$ represent an arbitrary concept. The rough set of X in $\mathbb{S} = (U, A)$ with regard to a set of attributes $B \subseteq A$ is a pair $App_B(X) = (\underline{X}, \overline{X})$, where

$$\begin{aligned} \underline{X} &= \{u \in U : [u]_B \subseteq X\}, \\ \overline{X} &= \{u \in U : [u]_B \cap X \neq \emptyset\}. \end{aligned}$$

The sets $\underline{X}$ and $\overline{X}$ constructed for an attribute set $B \subseteq A$ are called B-lower and B-upper approximations and the pair $App_B(X) = (\underline{X}, \overline{X})$ is sometimes called a *B-rough set* of X in $\mathbb{S}$. However, when the set B is fixed (or irrelevant) we will call the sets $\underline{X}$ and $\overline{X}$ simply the lower and upper approximations of X.

Of course, since an indiscernibility class of any object in U is a definable set in $\mathbb{S}$, the definitions of a rough set by definable sets and indiscernibility classes are equivalent. The lower and upper approximations can also be defined in several other equivalent ways, which might be convenient when dealing with specific problems [118, 133]. The above definition makes it obvious that the lower approximation of a concept can be described using predecessors of consistent rules, whereas the description of the upper approximation may require some rules with the confidence factor lower than 1. This fact will be used during the construction of a similarity model proposed in Section 4.3.2.

For the classical definition of rough set and for any $B \subseteq A$, the lower and upper approximations of $X \subseteq U$ have several interesting properties:

(L1)	$\underline{X} \in DEF(\mathbb{S})$	(U1)	$\overline{X} \in DEF(\mathbb{S})$
(L2)	$X \in DEF(\mathbb{S}) \Rightarrow \underline{X} = X$	(U2)	$X \in DEF(\mathbb{S}) \Rightarrow \overline{X} = X$
(L3)	$\underline{X} \subseteq X$	(U3)	$X \subseteq \overline{X}$
(L4)	$\underline{X} = U \setminus \overline{(U \setminus X)}$	(U4)	$\overline{X} = U \setminus \underline{(U \setminus X)}$
(L5)	$\underline{(X \cap Y)} = \underline{X} \cap \underline{Y}$	(U5)	$\overline{(X \cup Y)} = \overline{X} \cup \overline{Y}$
(L6)	$\underline{(X \cup Y)} \supseteq \underline{X} \cup \underline{Y}$	(U6)	$\overline{(X \cap Y)} \subseteq \overline{X} \cap \overline{Y}$
(L7)	$X \subseteq Y \Rightarrow \underline{X} \subseteq \underline{Y}$	(U7)	$X \subseteq Y \Rightarrow \overline{X} \subseteq \overline{Y}$
(L8)	$\underline{X} = \underline{(\underline{X})}$	(U8)	$\overline{X} = \overline{(\overline{X})}$
(L9)	$\underline{X} = \overline{(\underline{X})}$	(U9)	$\overline{X} = \underline{(\overline{X})}$

where $App_B(X) = (\underline{X}, \overline{X})$. Proofs of those properties are omitted since they are quite obvious and have already been presented in rough set literature (e.g. [110]). The properties (L4) and (U4) show that the lower and upper approximations are, in a sense, dual operations. In general, the other properties with the same number may be

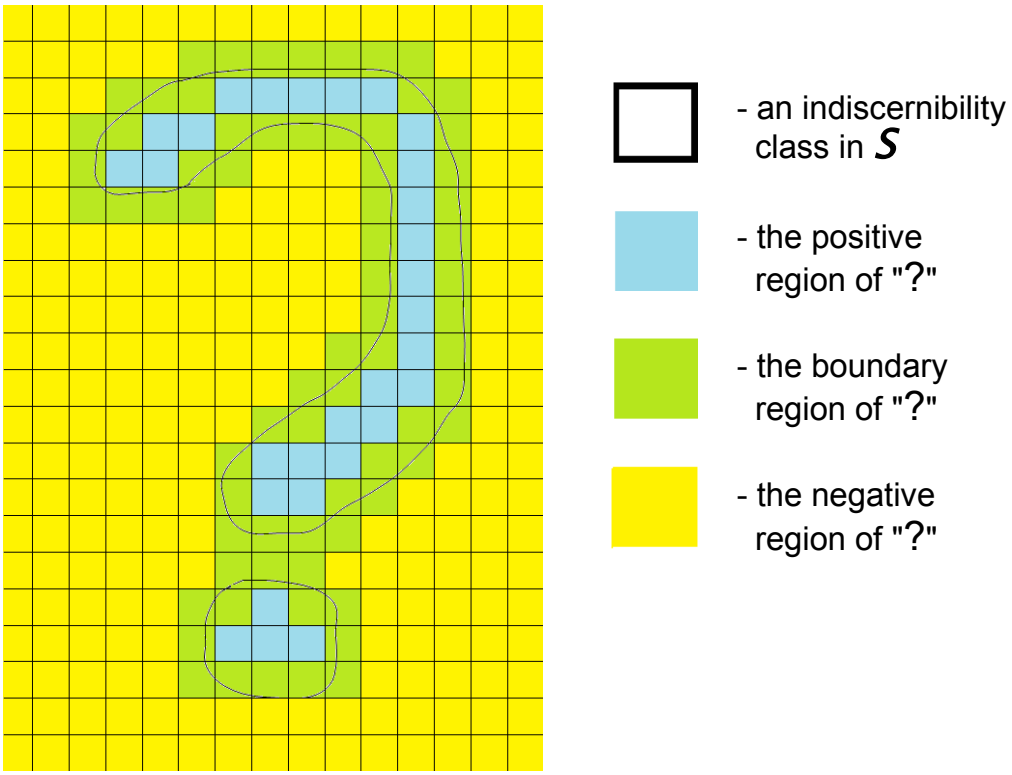


Figure 2.1: An exemplary rough set approximation of a concept.

regarded as dual. The properties (L1-2) and (U1-2) say that the two approximations are definable set (also called *crisp sets*). The properties (L3) and (U3) imply that for any set X, $\underline{X} \subseteq X \subseteq \overline{X}$. By the properties (L5-7) and (U5-7) it is shown that the operations of the lower and upper approximation are monotonic with regard to set inclusion, and the properties (L8-9), (U8-9) state that chains of rough set approximations are stable.

A B-rough set of a given set X defines a partitioning of objects from an information system into three disjoint sets called a *B-positive region*, *B-boundary region* and *B-negative region*. The positive region corresponds to the lower approximation of X - it contains objects that surely belong to the considered concept. It is usually denoted by $POS_B(X)$. The boundary region $BND_B(X)$ consists of objects whose belongingness is unclear (relative to a given set of attributes). It can be expressed as a difference between the upper and lower approximations: $BND_B(X) = \overline{X} - \underline{X}$. Finally, the negative region $NEG_B(X)$ contains objects that definitely do not belong to X, since they are outside its upper approximation: $NEG_B(X) = U \setminus \overline{X}$. Figure 2.1 shows rough set regions of an exemplary concept.

Zdzisław Pawlak in his early works on rough sets suggested an intuitive measure of rough approximation accuracy:

$$\alpha(App_B(X)) = \frac{|\underline{X}|}{|\overline{X}|} \quad .$$

The accuracy measure α expresses how well a given concept is modelled by its rough set. This measure is closely related to *roughness* of a set:

$$\rho_B(X) = \frac{|BND_B(X)|}{|\overline{X}|} = 1 - \alpha\big(App_B(X)\big) \ .$$

It is important to realize that the accuracy and roughness evaluate the rough approximations only on the available objects from an information system. Unfortunately, a close approximation on known data does not necessarily lead to a reliable assessment of new cases due to the over-fitting problem [93, 157]. However, those measures are still useful for tasks such as the feature selection, where they can help evaluating the impact of including or excluding an attribute from a given attribute set [94, 100, 174].

2.2.2 Approximation spaces

Although the rough set approximation of a concept is defined only for known objects from $\mathbb{S}$ it can be easily extended to all objects from Ω by considering descriptions of the lower and upper approximations. If the aim of the rough set analysis is to create a predictive model, then the quality of approximation on previously unseen cases is much more important than for the objects described in the decision table. To ensure this property, it is often necessary to modify representation of objects in the decision system by reducing unimportant or misleading attributes or by constructing new ones which are more informative. Such an operation influences the shape of the family of definable sets in $\mathbb{S}$, i.e. it changes the *approximation space* [133, 135] constructed for $\mathbb{S}$.

More formally, an approximation space is a tuple $\mathbb{A} = (U, IND)$, where U is a subset of known objects from Ω and $IND \subset U \times U$ is an indiscernibility relation [133]. This notion can be generalized by introducing two important concepts, namely an *uncertainty function* and a *f-membership function*.

Definition 2.5 (Uncertainty function)**.**
Let $U \subseteq \Omega$. A function $I : U \to \mathbb{P}(U)$ will be called an uncertainty function iff the following conditions are met:

1. $\forall_{u \in U} u \in I(u)$.
2. $u_1 \in I(u_2) \Leftrightarrow u_2 \in I(u_1)$.

The uncertainty function assigns neighbourhoods to objects from the set U. The conditions from Definition 2.5 imply that the uncertainty function defines a *tolerance relation*, i.e. a relation that is reflexive and symmetric [133]. However, in rough set literature this condition is sometimes weakened to consider any reflexive relation [102].

The sets defined by the uncertainty function may be utilized to measure a degree in which an object belongs to a given concept. It is usually done using an f-membership function.

Definition 2.6 (f-membership function)**.**
Let $U \subseteq \Omega$, $I : U \to \mathbb{P}(U)$ be an uncertainty function, $f : [0,1] \to [0,1]$ be a non-decreasing function and $\eta : U \times \mathbb{P}(U) \to \mathbb{R}$ be a function defined as:

$$\eta_I(u, X) = \frac{|I(u) \cap X|}{|I(u)|} \ .$$

A function $\mu = f(\eta)$ will be called an f-membership function.

If f is an identity function, then the f-membership function will be called simply a *membership function*. This type of an f-membership function coupled with a data driven uncertainty function will be explicitly used in the construction of the similarity model described in Section 4.3.

Having defined the uncertainty and the membership functions, a generalized approximation space can be defined as a tuple $\mathbb{A} = (U, I, \mu)$, where U is a subset of known objects from Ω, $I : U \to \mathbb{P}(U)$ is an uncertainty function and μ is an f-membership function.

In the classical rough set theory, the uncertainty function I often associates objects with their indiscernibility classes (i.e. $I(u) = I_B(u) = [u]_B$ for $B \subseteq A$) and the f-membership function has a form of $\mu(u, X) = \mu_B(u, X) = \frac{|[u]_B \cap X|}{|[u]_B|}$. For example, if we consider the information system from Table 2.1.a and the uncertainty function $I(u) = [u]_{\{a_1,a_2\}}$, the neighbourhood of u_2 would be $I(u_2) = \{u_2, u_4, u_7\}$. Furthermore, a degree to which u_2 belongs to the decision class with a label 1 with regard to I is equal to $\mu\big(u_2, \{d = 1\}(U)\big) = 2/3$.

In this way, the function I can be used to generalize the indiscernibility relation and define a new family of sets that can serve as building-blocks for constructing approximations. Coupled with the rough membership function, it leads to a more flexible definition of the lower and upper approximations:

$$\underline{X} = \{u \in U : \mu_I(u, X) = 1\} \ , \tag{2.7}$$

$$\overline{X} = \{u \in U : \mu_I(u, X) > 0\} \ . \tag{2.8}$$

Of course, if I is a description identity function, those definitions are equivalent to the classical ones. There also exist further generalizations of rough approximations, such as the variable precision rough set model [77, 175] which introduces an additional parameter allowing to weaken the zero-one bounds in the above definitions.

The uncertainty function can be defined, for example, by combining transformations of object representation space (the set of attributes) with the classical indiscernibility. Such a transformation may include reduction of the information describing objects to attributes which are truly related to the considered problem, as well as an extension of the attribute set by new, often higher-level features.

2.2.3 Approximation of relations

The rough approximations allow not only to express the uncertainty about concepts but also to model arbitrary relations between objects from Ω [132]. In fact, the notion of approximation spaces was generalized in [132] to allow defining approximations of sets in $\mathbb{U} = U_1 \times \ldots \times U_k$, where $U_i \subset \Omega$ are arbitrary sets of objects. Since the scope of this dissertation is on a similarity which can be seen as a binary relation (see Chapter 3), only this type of relations will be considered in this section.

A binary relation r between objects from a given set U is a subset of a Cartesian product of this set ($r \subseteq U \times U$). Having a subset of objects from Ω we may try to approximate an arbitrary binary relation $r \subseteq \Omega \times \Omega$ within the set $U \times U$ by considering a generalized approximation space, defined as a tuple $\mathbb{A}_2 = (U \times U, I_2, \mu_2)$, where $U \subset \Omega$, $I_2 : U \times U \to \mathbb{P}(U \times U)$ is a generalized uncertainty function and $\mu_2 : \big(U \times U\big) \times \big(\mathbb{P}(U \times U)\big) \to \mathbb{R}$ is a generalized rough membership function.

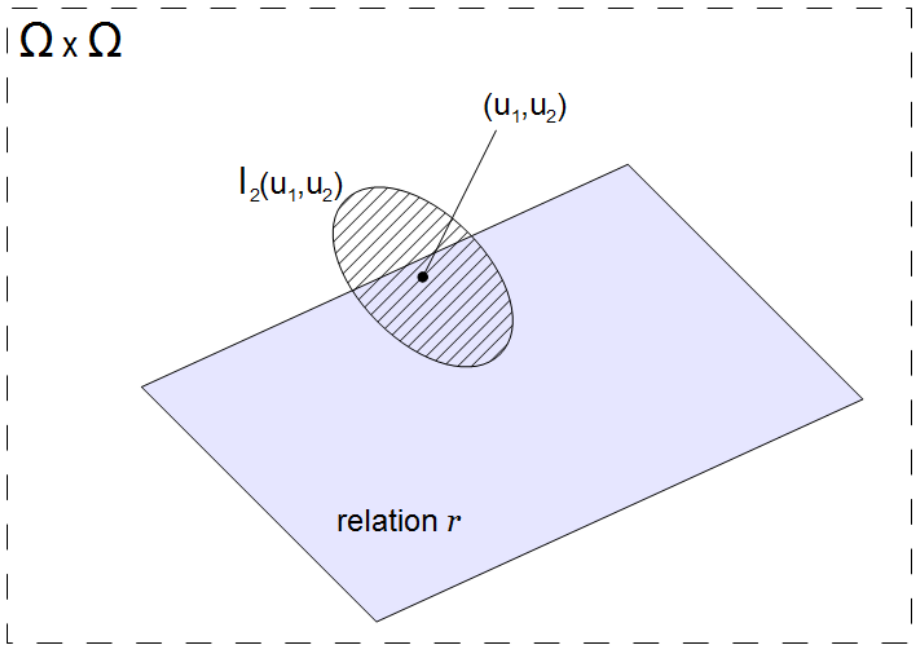


Figure 2.2: A graphical interpretation of as uncertainty function for approximation of a binary relation. In this case a membership function value $\mu_2\big((u_1, u_2), r\big)$ could be defined as a ratio between a size of the intersection of $I_2(u_1, u_2)$ and r, and the size of whole $I_2(u_1, u_2)$.

The functions I_2 and μ_2 can be easily defined by an analogy with the case of a regular approximation space. Their simplified graphical interpretation is shown in Figure 2.2. However, the meaning of an indiscernibility class of a pair of objects needs to be adjusted. In general, a pair (u_1, u_2) can be characterised by three possibly different sets of features – features specific to u_1, features specific to u_2 and those which describe u_1 and u_2 as a pair. This fact is utilized in a construction of the Rule-Based Similarity (RBS) model proposed in Section 4.3. In this model, objects are represented in a new feature space that allows for a robust approximation of a similarity relation. Such approximation is likely to be precise not only on training data but also in a situation when the model is used for assessment of resemblance of the training cases to completely new objects.

Approximations of a binary relation may have two desirable properties that indicate their quality, namely the *consistence* and *covering* properties defined below:

Definition 2.7 (Consistence property)**.**
Let $U \subseteq \Omega$ and r be a binary relation in Ω. We will say that a binary relation r' is consistent with r in U iff the implication

$$(u_1, u_2) \in r' \Rightarrow (u_1, u_2) \in r$$

holds for every $u_1, u_2 \in U$.

Definition 2.8 (Covering property)**.**
Let $U \subseteq \Omega$ and r be a binary relation in Ω. We will say that a binary relation r' covers r in U iff the implication

$$(u_1, u_2) \in r \Rightarrow (u_1, u_2) \in r'$$

holds for every $u_1, u_2 \in U$.

A fact that a relation r' is consistent with r in U will be denoted by $r' \subseteq_U r$. Analogically, a fact that r' covers r in U will be denoted by $r' \supseteq_U r$.

An approximation of a relation that has the consistence property can be seen as a kind of a rough set lower approximation, whereas an approximation that covers a binary relation can be treated as its upper approximation. Those two notions will be used in Chapter 3 to characterize a class of similarity functions that is the main scope of this dissertation.

2.3 Attribute Reduction

The problem of finding a representation of objects, which is appropriate in a given task, can be seen as a process of adaptation of an approximation space, therefore it is closely related to the rough sets in general. Zdzisław Pawlak wrote in [111] that discovering redundancy and dependencies between attributes is one of the fundamental and the most challenging problems of the rough set philosophy. The rough set theory provides intuitive tools for selecting informative features and constructing new ones. The most important of such tools are the notions of *information* and *decision reducts*.

2.3.1 Rough set information reduction

In many applications information about objects from a considered universe has to be reduced. This reduction is necessary in order to limit resources that are needed by algorithms analysing the data or to prevent crippling their performance by noisy or irrelevant attributes [50, 91, 93]. This vital problem has been in the scope of the rough set theory since its beginnings [110, 113, 115] and has been investigated by numerous researchers [69, 71, 94, 99, 100, 162].

Typically, in the rough set theory selecting compact yet informative sets of attributes is conducted using the notion of indiscernibility, by computing so called reducts [110, 131].

Definition 2.9 (Information reduct)**.**
Let $\mathbb{S} = (U, A)$ be an information system. A subset of attributes $IR \subseteq A$ will be called an information reduct iff the following two conditions are met:

1. *For any $u \in U$ the indiscernibility classes of u with regard to IR and A are equal, i.e. $[u]_A = [u]_{IR}$.*

2. *There is no proper subset $IR' \subsetneq IR$ for which the first condition holds.*

An information reduct IR can be interpreted as a set of attributes that are sufficient to discern among as many objects described in $\mathbb{S}$ as the whole attribute set A. At the same time the reduct is minimal, in a sense that no further attributes can be removed from IR without losing the full discernibility property. Analogically, it is possible to define a decision reduct DR for a decision system $\mathbb{S}_d$:

Definition 2.10 (Decision reduct)**.**
Let $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ be a decision system with a decision attribute d that indicates belongingness of objects to an investigated concept. A subset of attributes $DR \subseteq A$ will be called a decision reduct iff the following two conditions are met:

1. *For any $u \in U$ if the indiscernibility class of u relative to A is a subset of some decision class, its indiscernibility class relative to DR should also be a subset of that decision class, i.e. $[u]_A \subseteq [u]_d \Rightarrow [u]_{DR} \subseteq [u]_d$.*

2. *There is no proper subset $DR' \subsetneq DR$ for which the first condition holds.*

Unlike in the definition of information reducts, a decision reduct needs only to sustain the ability to discriminate objects from different decision classes. For example, $\{a_1, a_3, a_6\}$ and $\{a_3, a_5, a_6, a_7\}$ are information reducts of the information system from Table 2.1.a while $\{a_3, a_5\}$ and $\{a_3, a_6\}$ are decision reducts of the corresponding decision system.

The minimality of reducts stays in accordance with the Minimum Description Length (MDL) rule. Depending on an application, however, the minimality requirement for the reducts may sometimes be relaxed in order to ensure inclusion of the key attributes to the constructed model. In some cases keeping relevant but highly interdependent attributes may have a positive impact on model's performance [50, 91]. For this reason within the theory of rough sets a notion of decision superreduct is considered which is a set of attributes that discerns all objects from different decision classes but does not need to be minimal.

Usually for any information system there are numerous reducts. In the rough set literature there are described many algorithms for attribute reduction. The most commonly used are the methods utilizing discernibility matrices and the boolean reasoning [98, 100, 113, 131], and those which use a greedy or randomized search in the attribute space [64, 71, 137, 140].

In [110] it is shown that a decision reduct can consist only of strongly and weakly relevant attributes (it cannot contain any irrelevant attribute)[2] if the available data is sufficiently representative for the universe at scope. However, in real-life situations this requirement is rarely met. Very often, especially when analysing high dimensional data, some dependencies between attribute values and decisions are not general – they are specific only to a given data set. In such a case attributes which are in fact irrelevant might still be present in some decision reducts.

2.3.2 Generalizations of reducts

Many researchers have made attempts to tackle the problem of attribute relevance in decision reducts. Apart from devising heuristic algorithms for performing the

[2]The strong and weak relevance of attributes is understood as in [81].

attribute reduction that are more likely to select relevant features, a significant effort has been made in order to come up with some more general definitions of the reducts.

It has been noticed that subsets of attributes which preserve discernibility of a slightly lower number of objects from different decision classes than the whole attribute set tend to be much smaller than the regular reducts. Usually objects that are described with fewer attributes have larger discernibility classes which correspond to more general decision rules. This observation motivated introduction of the notion of an approximate decision reduct [96, 101, 140, 136, 138].

Definition 2.11 (Approximate decision reduct)**.**
Let $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ be a decision system with a decision attribute d and let ϵ be a real non-negative number, $\epsilon \in [0, 1)$. Additionally, let $|POS_B(d)|$ denote the number of objects whose indiscernibility classes with regard to an attribute set $B \subseteq A$ are subsets of a single decision class, i.e. $|POS_B(d)| = |\{u \in U : [u]_B \subseteq [u]_d\}|$. A subset of attributes $ADR \subseteq A$ will be called an ϵ-approximate decision reduct iff the following two conditions are met:

1. *ADR preserves discernibility in $\mathbb{S}_d$ with a degree of $1 - \epsilon$, i.e. $|POS_{ADR}(d)| \geq (1 - \epsilon) \cdot |POS_A(d)|$.*

2. *There is no proper subset $ADR' \subsetneq ADR$ for which the first condition holds.*

Of course, for $\epsilon = 0$ this definition is equivalent to the definition of regular decision reducts. For the decision system from Table 2.1.b the attribute subsets $\{a_1, a_3\}$ and $\{a_5, a_6\}$ are examples of the 0.3-approximate decision reducts.

The ϵ-approximate reducts can also be defined using differently formulated conditions. For example, instead of relying on the sizes of positive regions of decision classes, the approximate decision reducts can be defined based on the conditional entropy [138] of an attribute set or the number of discerned object pairs [96]. In fact, any measure of dependence between a conditional attribute subset and the decision, which is monotonic with regard to inclusion of new attributes, can be used [140].

A different generalization of the decision reducts, called dynamic decision reducts, has been proposed in [11]. In this approach a stability of a selected attribute set is additionally verified by checking if all the attributes are still necessary when only some smaller random subsets of objects are considered.

Definition 2.12 (Dynamic decision reduct)**.**
Let $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ be a decision system with a decision attribute d and let $RED(\mathbb{S}_d)$ be a family of all decision reducts of $\mathbb{S}_d$. Moreover, let ϵ and δ be real numbers such that $\epsilon, \delta \in [0, 1)$. A subset of attributes $DDR \subseteq A$ will be called an (ϵ, δ)-dynamic decision reduct iff for a finite set of all subsystems of $\mathbb{S}_d$, denoted by $SUB(\mathbb{S}_d, \epsilon)$, such that for each $\mathbb{S}'_d = (U', A, d) \in SUB(\mathbb{S}_d, \epsilon)$, $U' \subset U$ and $|U'| \leq (1 - \epsilon) \cdot |U|$, the following two conditions are met:

1. *DDR is a decision reduct of $\mathbb{S}_d$ ($DDR \in RED(\mathbb{S}_d)$).*

2. *DDR is a decision reduct of sufficiently many $\mathbb{S}'_d \in SUB(\mathbb{S}_d, \epsilon)$, i.e. $|\{\mathbb{S}'_d \in SUB(\mathbb{S}_d, \epsilon) : DDR \in RED(\mathbb{S}'_d)\}| \geq (1 - \delta) \cdot |SUB(\mathbb{S}_d, \epsilon)|$.*

Intuitively, if none of the attributes selected as belonging to a decision reduct is redundant when considering only subsets of objects, then the reduct can be seen as insensitive to data disturbances. Due to this characteristic the dynamic decision reducts are more likely to define robust decision rules [9, 11]. Additionally, the dynamic decision reducts tend to be more compact than the regular reducts. For example, from two decision reducts $DR_1 = \{a_3, a_5\}$ and $DR_2 = \{a_1, a_2, a_8\}$ of the decision system $\big(U, A \cup \{d\}\big)$ from Table 2.1.b, only the first one is a $(0.1, 0)$-dynamic decision reduct, since DR_2 is not a reduct of a decision system $\mathbb{S}'_d = (U \setminus \{u_4\}, A, d)$.

Both of those generalizations of the decision reducts have been successfully used in applications, such as constructing ensembles of predictive models [169], discovering of approximate dependencies between attributes [101, 140] and attribute ranking [71]. In this dissertation it is also showed how the dynamic reducts can be utilized for learning of a similarity function [65, 67] (see also Chapter 4). The definitions of approximate and dynamic reducts for information systems can be given analogously to those for the decision systems, thus they are omitted.

2.3.3 Notion of bireducts

The original definition of a decision reduct is quite restrictive, requiring that it should provide the same level of information about decisions as the complete set of available attributes. On the other hand, the approximate reducts, which are usually smaller and provide a more reliable basis for constructing classifiers [140, 144], can be defined in so many ways that selecting the optimal one for a given task is very difficult. The choice of the method may depend on a nature of particular data sets and on a purpose for the attribute reduction. Moreover, computation of the approximate decision reducts may require tuning of some unintuitive parameters, such as the threshold for a stopping criterion (ϵ).

Another issue with the approximate reducts is related to the problem of building classifier ensembles [5, 30, 144, 151, 169]. Combining multiple classifiers is efficient only if particular models tend to make errors on different areas of the universe at scope. Although, in general, there is no computationally feasible solution that can guarantee such a diversity, several heuristic approaches exist. For instance, one may focus on the classifier ensembles learnt from reducts that include as different attributes as possible. In this way one may increase stability of the classification and improve the ability to represent data dependencies to the users. Unfortunately, the common approximate reduct computation methods do not provide any means for controlling which parts of data are problematic for particular reducts. As a result, when building an ensemble where individual reducts are supposed to correctly classify at least 90% of the training objects, we may fail to anticipate that each of the resulting classifiers will have problems with the same 10% of instances.

To tackle the above challenges, a new extension of the original notion of a reduct was proposed [70, 141], called a *decision bireduct*. In this approach the emphasis is on both, a subset of attributes that describes the decision classes and a subset of objects for which such a description is possible.

Definition 2.13 (Decision bireduct)**.**
Let $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ *be a decision system. A pair* (B, X)*, where* $B \subseteq A$ *and* $X \subseteq U$*,*

is called a decision bireduct, iff B is a decision reduct of a subsystem (X, A, d) and the following properties hold:

1. *B discerns all pairs of objects from different decision classes in X and there is no proper subset $C \subsetneq B$ for which such a condition is met.*
2. *There is no $Y \supsetneq X$ such that B discerns all pairs of objects from different decision classes in (Y, B, d).*

It is important to realize that a decision subsystem (X, B, d) is always consistent (all indiscernibility classes in (X, B, d) are subsets of the decision classes), regardless of the consistency of the original system. However, a decision bireduct (B, X) can be regarded as an inexact functional dependence in $\mathbb{S}_d$ linking the subset of attributes B with the decision d, just as in a case of approximate reducts. The objects in X can be used to construct a classifier based on B and the objects from $U \setminus X$ can be treated as outliers. The computation of bireducts can be seen as searching for an approximation space that allows to generate meaningful decision rules. Such rules are local, since they are defined only for objects from X. However, by neglecting the potentially noisy outliers, the rules induced from the decision bireducts (e.g. by considering the indiscernibility classes of objects from X) are more likely to be robust [141]. It has been noted that bireduct-based ensembles tend to cover much broader areas of data than the regular reducts, which leads to better performance in classification problems [141].

Chapter 3

Notion of Similarity

The notion of similarity has been in a scope of interest for many decades [41, 42, 51, 147]. Knowing how to discriminate similar cases (or objects) from those which are dissimilar in a context of a decision class would enable us to conduct an accurate classification and to detect unusual situations or behaviours. Although human mind is capable of effortless assessing the resemblance of even very complex objects [58, 127], mathematicians, computer scientists, philosophers and psychologist have not come up with a single methodology of building similarity models appropriate for a wide range of complex object classes or domains.

A variety of methods were used in order to construct such models and define a relation which would combine an intuitive structure with a good predictive power. Among those a huge share was based on some distance measures. In that approach objects are treated as points in a metric space of their attributes and the similarity is a decreasing function of the distance between them. Objects are regarded as similar if they are close enough in this space. Such models may be generalized by introducing a list of parameters to the similarity function, e.g. weights of attributes. Tuning them results in the relation better fitting to a dataset. Algorithms for computationally efficient optimization of parameters for common similarity measures in the context of information systems were studied in, for instance, [102, 149, 171].

One may argue that the relation of this kind is very intuitive because objects which have many similar values of attributes are likely to be similar. However, Amos Tversky [41, 159] showed in empirical studies that in some contexts similarity does not necessarily have features like symmetry or subadditivity which are implied by distance measures. This situation occurs particularly often when we compare objects of great complexity. The explanation for this may lie in the fact that complex objects can be similar in some aspects and dissimilar in others. A dependency between local and global similarities may be highly non-linear and in order to model it we need to learn this dependency from the data, often relying on the domain knowledge provided by an expert.

This chapter discusses general properties of the similarity understood as a binary relation between objects from a considered universe. The following Section 3.1 introduces the notion of a similarity relation and explains some difficulties related to the formal definition of this idea. In its last subsection (Section 3.1.4) it describes how a performance of a similarity model can be quantitatively evaluated. Next,

Section 3.2 briefly overviews the most commonly used approaches to the problem of modelling the similarity relation. Its main focus is on showing the differences between the distance-based model and the approach proposed by Amos Tversky [159, 160]. Finally, the last section in this chapter (Section 3.3) shows exemplary applications of the similarity in fields such as Case-Based Reasoning and Cluster Analysis.

3.1 Similarity as a Relation

The similarity can be treated as a binary relation τ between objects from a universe Ω. Importance of this relation is unquestionable. In fact, many philosophers and cognitivists believe that the similarity plays a fundamental role in a process of learning from examples as well as acquiring new knowledge in general [51, 58, 127, 158]. Unfortunately, even though a human mind is capable of assessing similarity of even complex objects with a little effort, the existing computational models of this relation have troubles with accurate measuring of the resemblance between objects.

3.1.1 Vagueness of a similarity relation

Numerous empirical studies of psychologists and cognitivists showed that human perception of similar objects depends heavily on external factors, such as available information, personal experience and a context [41, 42, 159]. As a consequence, properties of a similarity relation may vary depending on both the universe and the context in which it is considered (see, e.g. [42, 159]). The similarity relation can be characterized only for a specific task or a problem. For instance, when comparing a general appearance of people in the same age, the similarity relation is likely to have a property of the symmetry. However, in a case when we compare people of a different age this property would not necessarily hold (e.g. a son is more similar to his father than the opposite). In a general case even the most basic properties, such as the reflexivity, can be questioned [159]. Figure 3.1 shows a drawing from two different perspectives. It can either be similar to itself or dissimilar, depending on whether we decide to consider its perspective.

The subjective nature of a similarity assessment makes it impossible to perfectly reflect the similarity using a single model. Capturing personal preferences would require tailoring the model to individual users. This could be hypothetically possible only if some personalized data was available and it would require some form of an automatic learning method. Even though in many applications it is sufficient to model similarity assessments of an "average user", a model which is designed for a given task and which takes into account the considered context, have much better chances to accurately measure the resemblance than an a priori selected general-purpose model.

Additionally, due to the fact that it is impossible to determine any specific features of the similarity without fixing its context, if no domain knowledge is available, it may be treated as a vague concept. In order to model it, all properties of this relation have to be derived from information at hand. Such information can usually be represented in an information system. That is another argument motivating the need for development of algorithms for learning domain-specific similarity relations

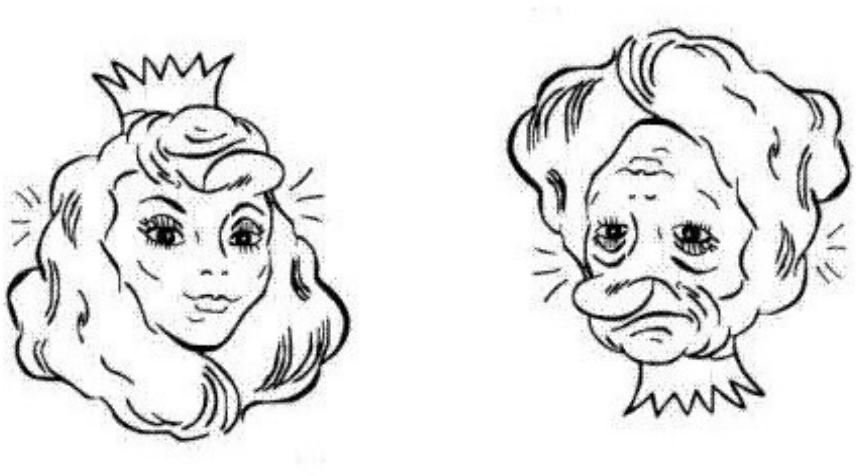


Figure 3.1: A single drawing from two different perspectives.

from data. One possible approach to this task is to utilize the theory of rough sets (see Chapter 2) to construct an approximation of τ, which will be denoted by τ^*.

Within the rough set theory, relations can be approximated just as any other concept (see Section 2.2.3). The problem of approximation of binary relations was investigated by researchers since the beginnings of the rough sets [112, 132, 145]. If no additional knowledge is available this task is much more difficult than, for instance, a classification. It may be regarded as a problem of assigning binary decision labels to pairs of instances from the universe Ω in an unsupervised manner, using information about a limited number of objects described in available information system. It is important to realize that the resulting approximation τ^* has to be reliable not only for objects at hand but also for new ones. For this reason, in practical situations, in order to properly approximate the similarity it is necessary to utilize some domain knowledge and to specify a context for the relation.

3.1.2 Similarity in a context

Several independent studies showed how important is to consider an appropriate context while judging a similarity between objects [42, 51, 147, 159]. Two stimuli presented to a representative group of people can be assessed as similar or dissimilar, depending on whether some additional information is given about their classification or whether they are shown along with some other characteristic objects. For instance, if we consider cars in a context of their class[1], then Chevrolet Camaro will be more similar to Ford Mustang than Ford Tempo. However, if we change the context to a make of a car the assessment would be completely different.

A selection of the context for the similarity has a great impact on features or in other words factors, that influence the judgements [42, 159]. In the previous example, a feature such as a colour of a car would be irrelevant in the context of car's class.

[1]The official classification of cars is discussed, e.g., in a Wikipedia article *Car classification* (http://en.wikipedia.org/wiki/Car_classification).

Nevertheless, it might be important in the context of a make of a car, since some car paints could be exclusively used by specific car producers.

When constructing a similarity model for a given data, the context for the relation can usually be inferred from a purpose which motivates performing the analysis. If a task is to cluster the given data into subsets of closely related objects in an unsupervised way and without any additional knowledge, then the context will probably be a general appearance of objects. However, if we know that, for example, the data describe textual documents, it is possible to consider them in a context of their semantics (their meaning – see Section 5.3). Furthermore, if the similarity model is created for a task such as a diagnosis of a specific condition based on a genetic profile of tissue samples, then a classification into severity stages of the condition will probably be the best context to choose (see experiments in Sections 5.1 and 5.2). In the last case the information specifying the context will usually correspond to a decision attribute in the data table.

It is also reasonable to consider similarity of two objects in a context of other objects in the data. For instance, a banana will be more similar to a cherry when considered in a data set describing dairy and meat products, vegetables and fruits, than in a case when the data is related only to different kinds of fruits. In those two cases, different aspects of the similarity would have to be taken into account, and as a consequence, the same attributes of the fruits would have different importance.

In general, similar objects are expected to have similar properties with regard to the considered context. Since in this dissertation the main focus is on the similarity in the context of a classification, the above principle can be reformulated in terms of the consistency of the similarity with the decision classes of objects[2]. More formally:

Definition 3.1 (Consistency with a classification)**.**
Let Ω be a universe of considered objects and let d be a decision attribute which can be used to divide objects from Ω into indiscernibility classes $\Omega/\{d\}$. Moreover, let τ denote a binary relation in Ω. We will say that τ is consistent with the classification indicated by d iff the following implication holds for every $u_1, u_2 \in \Omega$:

$$(u_1, u_2) \in \tau \Rightarrow d(u_1) = d(u_2) \, .$$

The above property will be referred to as the main feature of the similarity for the classification. It is also often assumed that a similarity relation in the context of a classification needs to be reflexive, namely $\forall_{u \in \Omega}(u, u) \in \tau$. Additionally, for objects which are described by a set of conditional attributes A, the reflexivity is understood in terms of indescernibility. In particular, we will say that τ is reflexive if and only if $\forall_{(u,u') \in IND_A}(u, u') \in \tau$. This assumption, however, can be true only if there are no two objects in Ω which are identical in all aspects but belong to different decision classes. Binary relations in Ω that have the above two properties will be regarded as *possible similarity relations* in the context of the classification. The set of all such relations will be denoted by:

$$R = \{\tau : \tau \subseteq_\Omega IND_{\{d\}} \wedge \tau \supseteq_\Omega IND_A\} \, .$$

In the remaining parts of the dissertation it is assumed that one of such relations $\tau \in R$ is fixed and considered as the reference similarity relation in the specified

[2]The consistency of two relations within a given set is defined in Section 2.2.3

classification context. It should be noted, however, that different scenarios for inducing this relation from data may or may not assume the availability of knowledge regarding τ for the training data. In the second case, which applies to the similarity model proposed in Section 4.3, only information about the properties of τ is utilized in the learning process.

The property from Definition 3.1 can be used to guide the process of constructing approximations of the relation τ. It infers that a desirable approximation τ^* should also be consistent with the decision classes indicated by d. In practice, however, this condition can be verified only for the known objects described in a decision system. Moreover, in real life applications it may sometimes be slightly relaxed in order to increase the recall of the approximation. Nevertheless, the knowledge that any set of objects that are similar to a given one must have the same decision can be used to limit a search space for features that can conveniently represent pairs of objects in an approximation space, as discussed in Section 2.2.3. It is also the fundamental assumption used in the construction of the Rule-Based Similarity in Chapter 4.

Although an approximation of a similarity in a context of classification can be made only using known objects from a given decision system, it has to allow an assessment of whether an arbitrary object from $\Omega \setminus U$ is similar to an object from U. To make this possible, an assumption is made that for objects from $\Omega \setminus U$ we can retrieve values of their conditional attributes (without the need for referring to their decision class, which may remain unknown).

There can be many approximations of a similarity relation for a given decision table $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$. For example, one can always define a trivial approximation for which no pair of objects is similar or a naive one, for which only objects from U that are known to belong to the same decision class can be similar. Therefore, in practical applications it is crucial to have means to evaluate quality of an approximation and estimate how close it is to the real similarity for the objects that are not described in $\mathbb{S}_d$, i.e. $\{u' \in \Omega : u' \notin U\}$. Since there is no available information regarding those objects, the Minimum Description Length rule (MDL) is often used to select the approximation which can be simply characterized but is sufficiently precise.

3.1.3 Similarity function and classification rules

In a variety of practical situations it is convenient to express a degree in which one object is similar to another. For instance, in machine learning many classification methods construct rankings of training objects based on their similarity to a considered test case (e.g. the *k nearest neighbours* algorithm [15, 93, 102, 166]).

To assess the level of similarity between a pair of objects, a special kind of function is used, called a *similarity function*. Usually, such a function for a considered data set is given a priori by an expert for a whole $\Omega \times \Omega$ set, independently of the available data and the context. Intuitively, however, a similarity function for an information system $\mathbb{S} = (U, A)$ should be a function $Sim : U \times \Omega \to \mathbb{R}$, whose values are "high" for objects in a true similarity relation and becomes "low" for objects not in this relation. Such a function could be used to define a family of approximations of the similarity relation τ by considering the sets $\tau_{(\lambda)}^{Sim} = \{(u_1, u_2) \in U \times U : Sim(u_1, u_2) \geq \lambda\}$ for any $\lambda \in \mathbb{R}$. If a function Sim is appropriate for a given relation, then at least some of

Table 3.1: A similarity relation in a context of classification for the objects from the decision system depicted in Table 2.1.b (on the left) and a table with the corresponding values of a similarity function (on the right).

$$\tau = \big\{(u_1,u_1),(u_1,u_7),(u_2,u_2),(u_2,u_5),(u_2,u_7),(u_3,u_3),(u_3,u_7),(u_4,u_4),(u_4,u_8),(u_5,u_2),(u_5,u_5),(u_5,u_7),(u_6,u_6),(u_6,u_8),(u_7,u_1),(u_7,u_2),(u_7,u_5),(u_7,u_7),(u_8,u_4),(u_8,u_8),(u_9,u_4),(u_9,u_9)\big\}$$

A similarity matrix:

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
u_1	1.00	0.50	0.42	0.09	0.51	0.23	0.66	0.09	0.09
u_2	0.50	1.00	0.42	0.09	1.00	0.20	0.67	0.08	0.43
u_3	0.42	0.42	1.00	0.00	0.42	0.00	0.62	0.00	0.32
u_4	0.07	0.09	0.00	1.00	0.08	0.50	0.00	1.00	0.48
u_5	0.50	1.00	0.42	0.09	1.00	0.20	0.66	0.08	0.12
u_6	0.20	0.25	0.00	0.50	0.20	1.00	0.09	0.50	0.20
u_7	0.68	0.66	0.60	0.00	0.66	0.07	1.00	0.00	0.00
u_8	0.09	0.09	0.00	1.00	0.12	0.50	0.00	1.00	0.33
u_9	0.09	0.50	0.32	0.50	0.11	0.20	0.00	0.33	1.00

the approximations $\tau^{Sim}_{(\lambda)}$ should be consistent with τ (see Definition 2.7) for available data. To further formalize this notion for the purpose of this dissertation a concept of a *proper similarity function* is proposed:

Definition 3.2 (Proper similarity function)**.**
Let τ be a similarity relation between objects from Ω, $U \subseteq \Omega$ be a subset of known reference objects and $Sim : U \times \Omega \to \mathbb{R}$ be a function. We will say that Sim is a proper similarity function for the relation τ within the set U iff there exist $\epsilon_1, \epsilon_2 \in \mathbb{R}$, $\epsilon_1 > \epsilon_2$, such that the following conditions hold:

1. $\left|\tau^{Sim}_{(\epsilon_1)}\right| > 0$ *and* $\tau^{Sim}_{(\epsilon_1)} \subseteq_U \tau$ *(see Def. 2.7),*
2. $\left|(U \times U) \setminus \tau^{Sim}_{(\epsilon_2)}\right| > 0$ *and* $\tau^{Sim}_{(\epsilon_2)} \supseteq_U \tau$ *(see Def. 2.8).*

A value of a similarity function for a pair (u_1, u_2) will be called *a similarity degree* of u_1 relative to u_2. Each of the sets $\tau^{Sim}_{(\lambda)}$ can be regarded as an approximation of the similarity relation τ. The first condition from Definition 3.2 requires that, starting from some ϵ_1, all the approximations $\tau^{Sim}_{(\lambda)}$ defined by a proper similarity function were subsets of the true similarity relation. It means that the precision of the approximation defined as $prec_\tau(\tau^{Sim}_{(\lambda)}) = \frac{|\tau^{Sim}_{(\lambda)} \cap \tau|}{|\tau^{Sim}_{(\lambda)}|}$ equals 1 for all $\lambda \geq \epsilon_1$ such that $|\tau^{Sim}_{(\lambda)}| > 0$. One practical implication of this fact is that in the context of classification, for sufficiently large λ, objects in each pair from $\tau^{Sim}_{(\lambda)}$ must belong to the same decision class (see Definition 3.1).

The second condition in the definition of a proper similarity function requires that there exists a border value ϵ_2 such that all $\tau^{Sim}_{(\lambda)}$ for $\lambda \leq \epsilon_2$ were supersets of the true similarity relation τ. By an analogy to the rough sets, $\tau^{Sim}_{(\epsilon_1)}$ and $\tau^{Sim}_{(\epsilon_2)}$ can be treated as a lower and upper approximation of the similarity, respectively (see Section 2.2).

Of course, one function can be a proper similarity function in one context but not in the other. If a function Sim is a proper similarity function for a given similarity

relation, we will say that this relation is approximable by Sim. For example, Table 3.1 shows a similarity relation between objects described in the decision system from Table 2.1.b and a similarity matrix displaying values of some similarity function for all the pairs of the objects. As one can easily notice, the relation τ is consistent with the decision classes. Moreover, the similarity function used to generate the matrix is a proper similarity function for τ within the considered set of objects because for $\lambda = 0.66$ the corresponding approximation $\tau^{Sim}_{(0.66)} = \big\{(u_1,u_1),(u_1,u_7),(u_2,u_2),(u_2,u_5),(u_2,u_7),(u_3,u_3),(u_4,u_4),(u_4,u_8),(u_5,u_2),(u_5,u_5),(u_5,u_7),(u_6,u_6),(u_7,u_1),(u_7,u_2),(u_7,u_5),(u_7,u_7),(u_8,u_4),(u_8,u_8),(u_9,u_9)\big\}$ is consistent with τ and for $\lambda = 0.50$ the approximation $\tau^{Sim}_{(0.50)} = \big\{(u_1,u_1),(u_1,u_2),(u_1,u_5),(u_1,u_7),(u_2,u_1),(u_2,u_2),(u_2,u_5),(u_2,u_7),(u_3,u_3),(u_3,u_7),(u_4,u_4),(u_4,u_6),(u_4,u_8),(u_5,u_1),(u_5,u_2),(u_5,u_5),(u_5,u_7),(u_6,u_4),(u_6,u_6),(u_6,u_8),(u_7,u_1),(u_7,u_2),(u_7,u_3),(u_7,u_5),(u_7,u_7),(u_8,u_4),(u_8,u_6),(u_8,u_8),(u_9,u_2),(u_9,u_4),(u_9,u_9)\big\}$ covers the relation τ.

Is is worth to notice that a proper similarity function does not need to be symmetric. For instance, in the previous example $Sim(u_1,u_4) \neq Sim(u_4,u_1)$. It is also important to realize that a similarity function does not need to be non-negative. The negative values of a similarity function are usually interpreted as an indication that the compared objects are more dissimilar than they are similar. However, the majority of commonly used similarity functions are non-negative.

A similarity function allows to order objects from U according to their degree of similarity to any given object from the considered universe. It is important to notice, that the similarity function allows to compute the similarity coefficient of u from the set of known objects U to any object from the universe Ω, given that it is possible to determine its attribute values. In particular, information about a decision class of the second object does not need to be available. That property may be used to define several simple, case-based classification methods. For instance, if the available training objects are described in a decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$, an object $y \in \Omega$ can be assigned to a decision class of the most similar object from U:

$$\text{1-NN}_{Sim}(y) = d\Big(\operatorname*{argmax}_{u \in U} Sim(u,y)\Big)\,. \tag{3.1}$$

This formula can be easily generalized to a k-nearest neighbours classification by introducing a voting scheme for deciding a class of the investigated case [23, 93, 107, 157, 166]. A voting scheme can also be applied in a case when in U there are several objects which are equally similar to y and belong to different decision classes. There are numerous voting schemes that aim at optimizing the classification performance. A basic heuristic is a majority voting by the k most similar objects from the system $\mathbb{S}_d$. Some more complex voting schemes may additionally take into account the actual similarity function values to weight the votes of the neighbours. The relative "importance" of a vote may also be adjusted by considering empirical probabilities of the decision classes.

A similarity function may also be used to define a slightly different kind of a classification rule. In the λ-majority classification, an object $y \in \Omega$ is assigned to a decision class which is the most frequent within the set of objects regarded as similar to y. Particularly, if we denote $C_\lambda(y) = \big\{u \in U : Sim(u,y) \geq \lambda\big\}$, then y can be classified as belonging to one of the l decision classes $d_1, ..., d_l$ of d using the formula:

$$\lambda\text{-majority}_{Sim}(y) = \operatorname*{argmax}_{d_j \in \{d_1,\ldots,d_l\}} \big|\{u \in U : u \in C_\lambda(y) \wedge d(u) = d_j\}\big|\,. \tag{3.2}$$

The λ-majority classification assigns objects to a class with the highest number of similar examples, according to an approximation of the similarity relation by the set $\tau_{(\lambda)}^{Sim}$. It also can make use of different voting schemes, such as object weighting by a similarity degree or considering sizes of the decision classes. Some exemplary similarity functions and their applications in the context of the classification task are discussed in Section 3.3.1.

The ability to assess a similarity degree is also useful in an unsupervised data analysis (see Section 3.3.3). For instance, various similarity functions are commonly used by clustering algorithms to form homogeneous groups of objects. Moreover, similarity functions may be more convenient to use for an evaluation of a similarity model, since the implicit verification of a similarity relation approximation may require checking all pairs of objects. More application examples of similarity functions for supervised and unsupervised learning are discussed in Section 3.3.

3.1.4 Evaluation of similarity models

A similarity relation in a given context can be approximated using many different methods. However, a quality of two different approximations will rarely be the same. In order to be able to select the one which is appropriate for a considered problem there have to be defined some means of measuring a compliance of the approximation with the real similarity relation.

An objective evaluation of similarity assessment is a problem that has always accompanied research on similarity models. Although there have been developed many methods for measuring the quality of a similarity model, the most of them can be grouped into three categories. The main criteria for this division is a required involvement of human experts.

In the first category there are methods which measure compliance of the assessment returned by a model with human-made similarity ratings. Such an approach includes researches in which human subjects are asked to assess the similarity between pairs of objects (called stimuli). Next, those assessments are compared with an output of the tested model and some statistics measuring their correspondence are computed. For instance, Tversky in [159] describes a study in which people were asked about a similarity between particular pairs of countries. As a part of this study, two independent groups of participants had to assess the similarity degrees between the same pairs of countries, but with an inverse ordering (i.e. one group assessed how similar is country A to B, whereas the second judged the similarity of B to A). Based on those ratings, Tversky showed that there is a statistically significant asymmetry in the average similarity judgements within those two groups and used this finding as an argument for viability of his feature contrast model (see Section 3.2.2). In a different study on the similarity of vehicles [159], Tversky measured the correlation between the average assessments made by human subjects and the results of his model. In this way he was able to show that taking into account both common and distinctive features of objects, his model can better fit the data than in a case when those sets of characteristics are considered separately.

The main advantage of this approach is that it allows to directly assess the viability of the tested model to a given problem. Average assessments made by human subjects

define the ground truth similarity relation which the model tries to approximate. By using well-defined statistical measures of compliance between two sets of judgements it is possible not only to objectively evaluate the model but also to quantitatively compare it to different models and decide which one is better.

However, such a direct approach has some serious disadvantages. It usually requires a lot of time and resources to gather a meaningful amount of data from human participants. This does not only increase the overall cost of the model but also limits the possible test applications to relatively small data sets. Additionally, it is sometimes difficult to design an environment for manual assessment of the similarity in a desired context. Since there are many factors that can influence human judgement, the similarity ratings obtained in this way can be biased. Due to those practical reasons, usage of this evaluation method is very rare for data sets with more than a few hundreds of stimuli.

Table 3.2: Summary of typical similarity model evaluation methods.

Correlation with average similarity ratings	
Advantages:	Disadvantages:
– direct assessment of a model – simple and intuitive evaluation	– requires human-made ratings – deficiencies in data availability – possibility of a context bias
Measures of compliance with constraints	
Advantages:	Disadvantages:
– semi-direct model assessment – simpler for experts	– requires experts to impose constraints by labelling or grouping – possible inconsistencies
Measures of classification accuracy	
Advantages:	Disadvantages:
– no human involvement required – no limitations on data availability or quality – applicable for large data sets	– indirect model assessment – can be used only in the context of classification

The second category of similarity model evaluation methods consists of measures that verify compliance of the tested model with constraints imposed by domain experts. Usually, even when a data set is too large to evaluate similarity degrees between every pair of objects, experts are able to define some rules that must be satisfied by a good similarity model. Such rules may be either very general (e.g. *less complex objects should be more similar to the more complex ones than the opposite*) or very specific (e.g. *object* u_1 *and* u_2 *must not be indicated as similar*). The quality of a model is then expressed as a function of a cardinality of a set of violated rules.

Experts may also provide some feedback regarding truly relevant characteristics of some objects in the considered context. This information can be utilized to heuristically assess the similarity degree of the preselected objects and those values may be used as a reference during the evaluation of similarity models. In a more general setting, this type of quality assessment can be used to measure quality

in a semantic clustering task [70] and motivates the semi-supervised clustering algorithms [4]. This approach is used in experiments described in Section 5.3 to evaluate the similarity models for scientific articles, constructed in the context of their semantic similarity.

The main advantage of this approach is that it is usually much more convenient for experts to specify constraints rather than indicate exact similarity values. Since such rules may be local and do not need to cover all pairs of objects, they might be applied to evaluate a similarity model on a much larger data. One major drawback is the possible inconsistency within the constraints defined by different experts. Also the evaluation cost which is related to the employment of human experts cannot be neglected.

Finally, the last category consists of methods that can only be applied in the context of classification. Similarity models are often built in order to support decision making or to facilitate a prediction of classes of new objects. If a model is designed specifically for this purpose, it is reasonable to evaluate its performance by measuring the quality of predictions made with a use of similarity-based decision rules (see Definitions 3.1 and 3.2). Since the main feature of similarity in a context of classification (Definition 3.1) imposes a kind of a constraint on desired assessments of similarity, this approach can be seen as a special case of the methods from the second category. However it differs in that, it does not need the involvement of human experts.

The biggest advantage of this approach is the lack of restrictions on evaluation data availability. It makes it possible to automatically test a similarity model even on huge data sets, which makes the evaluation more reliable. Due to those practical reasons this particular method was used in many studies, including [60, 64, 65, 67, 89, 102, 149]. It was also used in experiments conducted for the purpose of this dissertation which are described in Sections 5.1 and 5.2. Table 3.2 summarizes the above discussion on the methods for evaluation of similarity models.

3.2 Commonly Used Similarity Models

This section overviews the most commonly used similarity models. The presented approaches differ in the constrains on the way they approximate the similarity relation. For instance, the distance-based models restrict the approximations to relations which are reflexive and symmetric. However, all the models discussed in this section have one property in common. They can be used to approximate the similarity in a way that is independent of a particular data domain or a context. For this reason the resulting approximations are often not optimal and expert knowledge is needed to decide whether it is worth to apply a selected model to a given problem.

3.2.1 Distance-based similarity modelling

The most commonly used in practical applications are the distance-based similarity models. A basic intuition behind this approach is that each object from a universe Ω can be mapped to some point in an attribute value vector space. It is assumed that

in this space there is a metric defined which allows to assess a distance between any two points. Such a metric will be called a *distance function* or a *distance measure*.

Definition 3.3 (Distance measure)**.**
Let Ω be a universe of objects and let $Dist : \Omega \times \Omega \to \mathbb{R}^+ \cup \{0\}$ be a non-negative real function. We will say that $Dist$ is a distance measure if the following conditions are met for all $u_1, u_2, u_3 \in \Omega$:

1. $Dist(u_1, u_2) = 0 \Leftrightarrow u_1 = u_2$ *(identity of indiscernibles),*
2. $Dist(u_1, u_2) = Dist(u_2, u_1)$ *(symmetry),*
3. $Dist(u_1, u_2) + Dist(u_2, u_3) \geq Dist(u_1, u_3)$ *(triangle inequality).*

If the objects from Ω are described by attributes from a set A, then the first condition can be generalized by considering the indiscernibility classes of u_1 and u_2: $Dist(u_1, u_2) = 0 \Leftrightarrow (u_1, u_2) \in IND_A$. This particular variation of the distance measure definition will be used in the later sections. Moreover, if a given function does not fulfill the third condition (the triangle inequality) but meets the other two it is called a *semidistance* or a *semimetric*.

A typical example of a distance measure is the Euclidean distance, which is a standard metric in Euclidean spaces:

$$Dist_E(u_1, u_2) = \sqrt{\sum_{a \in A} \big(a(u_1) - a(u_2)\big)^2}. \tag{3.3}$$

Another example of a useful distance measure is the Manhattan distance:

$$Dist_M(u_1, u_2) = \sum_{a \in A} \big|a(u_1) - a(u_2)\big|. \tag{3.4}$$

Both of the above metrics are generalized by the Minkowski distances, which can be regarded as a parametrized family of distance measures:

$$Dist_p(u_1, u_2) = \Big(\sum_{a \in A} \big|a(u_1) - a(u_2)\big|^p \Big)^{1/p}. \tag{3.5}$$

Figure 3.2 presents shapes of circles in spaces with Minkowski metric for different values of the parameter p.

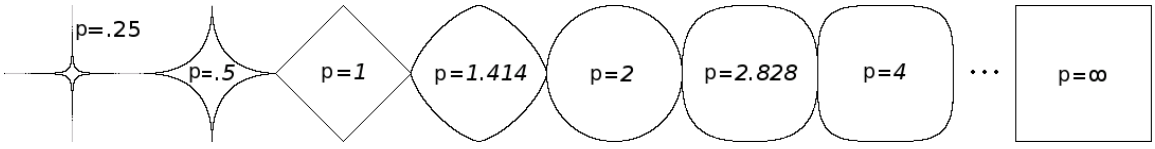


Figure 3.2: Shapes of circles in spaces with different Minkowski distances.

A different example of an interesting distance function in a $\mathbb{R}^{|A|}$ space is the Canberra distance:

$$Dist_C(u_1, u_2) = \sum_{a \in A} \frac{|a(u_1) - a(u_2)|}{|a(u_1)| + |a(u_2)|}. \tag{3.6}$$

It is mostly used for data with non-negative attribute values scattered around the centre since it has a property that its value becomes unity when the attributes are of opposite sign.

All the above metrics work only for objects described by numeric attributes. There are however numerous metrics which can be applied to cases with symbolic attributes. The most basic of those is the Hamming distance:

$$Dist_H(u_1, u_2) = |\{a \in A : a(u_1) \neq a(u_2)\}|. \tag{3.7}$$

Typically, the Hamming distance is used for the assessment of a proximity between binary strings. It can also be utilized for comparison of any equally sized strings, but in such a case the edit distance[3] is more commonly employed, since it allows to compare strings of different length.

Another example of a distance defined for objects with binary attributes is the binary distance:

$$Dist_b(u_1, u_2) = \frac{|\{a \in A : a(u_1) \neq a(u_2)\}|}{|\{a \in A : a(u_1) \neq 0 \vee a(u_2) \neq 0\}|}. \tag{3.8}$$

The binary distance can be applied to any type of symbolic data after transformation of each symbolic attribute to its binary representation.

A common choice of a measure in high dimensional numeric spaces is the cosine distance. It measures the angle between two vectors:

$$Dist_{arcc}(u_1, u_2) = \arccos(\cos(u_1, u_2)) \tag{3.9}$$

$$= \arccos\left(\frac{\sum\limits_{a\in A|} a(u_1) \cdot a(u_2)}{\sqrt{\sum\limits_{a\in A} \big(a(u_1)\big)^2} \cdot \sqrt{\sum\limits_{a\in A} \big(a(u_2)\big)^2}}\right). \tag{3.10}$$

The cosine between two vectors is equivalent to their scalar product divided by a product of their norms. A distance defined in this way is a proper metric only for points from $(\mathbb{R}^+)^{|A|}$ and lying on a sphere. To avoid computation of the arc-cosine, in applications this distance function is simplified to a form:

$$Dist_c(u_1, u_2) = 1 - \frac{\sum\limits_{a\in A|} a(u_1) \cdot a(u_2)}{\sqrt{\sum\limits_{a\in A} \big(a(u_1)\big)^2} \cdot \sqrt{\sum\limits_{a\in A} \big(a(u_2)\big)^2}}. \tag{3.11}$$

It needs to be noted, however, that $Dist_c$ is only a semimetric.

The main advantage of the cosine distance is that it can be efficiently computed even for extremely high dimensional but sparse data[4]. In such a case, representations of all objects can be normalized by dividing all attribute values by a norm of the corresponding vectors in Euclidean metric space. After this transformation, the distance can be computed by multiplying only attributes with non-zero values for

[3] A value of the edit distance is equal to the minimal number of edit operations needed to transform one string into another. It is often called the Levenshtein distance.

[4] Sparse data is data with little non-zero attribute values.

both points and summing the results. For this reason the cosine distance is commonly used in information retrieval [54, 122] and textual data mining [35, 55, 95].

It can be easily noted that the most of the above distance measures can be seen as a composition of two functions. The first one is applied independently for each of the attributes to measure how different their values are in the compared objects. The second one aggregates those measurements and expresses the final distance. For example, in a case of the Minkowski distance the first function is $f(x, y) = |x - y|$ and the second is $F_p(x_1, \ldots, x_{|A|}) = \big(\sum_{i=1,\ldots,|A|} x_i^p \big)^{1/p}$. Such functions are called a *local distance* and a *global distance*, respectively. It can be shown that a large share of distance measures can be constructed by composing a distance in a one-dimensional space and a norm in a $|A|$-dimensional vector space (a proof of this fact can be found e.g. in [59]). This fact is often called *a local-global principle*.

By applying different local distance types to attributes, it is possible to measure distances between objects described by a mixture of numerical and nominal features. One example of such a measure is the Gower distance. It uses the absolute value of difference and the equivalence test for numerical and nominal attributes, respectively, and then it aggregates the local distances using the standard Euclidean norm.

In the distance-based approach, a similarity is a non-increasing function of a distance between representations of two objects. The transformation from a distance to a similarity values is usually done using some simple monotonic function such as the linear transform (Equation 3.12) or the inverse transform (Equation 3.13). Many other functions, such as common kernels, can also be used.

$$Sim_{lin}(u_1, u_2) = C - Dist(u_1, u_2) \tag{3.12}$$

$$Sim_{inv}(u_1, u_2) = \frac{1}{Dist(u_1, u_2) + C} \tag{3.13}$$

In the above equations C is a constant, which is used to place the similarity values into appropriate interval. Some other scaling methods can sometimes be additionally applied to secure that the similarity values stay in a desired range for pairs of objects from a given information system.

The usage of distance measures for computation of a similarity makes the resulting model inherit some of the properties of metrics. For instance, any distance-based approximation of the similarity will always have the property of reflexivity and symmetry, which might be undesirable. Moreover, if a similarity function is based on a globally predefined distance measure, it does not take into account the influence of particular characteristics of objects in a given context and treats all the attributes alike. The distinction between the local and global distances makes it possible to partially overcome this issue by introducing additional parameters which express the importance of the local factors to the global similarity. One example of such similarity measure is based on the generalized Minkowski distance:

$$Dist_{w,p}(u_1, u_2) = \Big(\sum_{a \in A} w_a \cdot \big| a(u_1) - a(u_2) \big|^p \Big)^{1/p}. \tag{3.14}$$

In this model, the vector of parameters $w = (w_{a_1}, \ldots, w_{a_{|A|}})$ can be set by domain experts or can be tuned using one of the similarity function learning techniques discussed in Section 4.2.

From the fact that any distance-based similarity approximation has to be reflexive, it follows that a distance-based similarity function can be a proper similarity function only in a case when the true similarity is also reflexive. In practical situations the similarity may not have this property. For instance, when it is considered in the context of classification and there are some inconsistencies in the data.

3.2.2 Feature contrast model

Although the distance-based similarity models were successfully applied in many domains to support a decision making or to discover groups of related objects (examples of such applications are given in Section 3.3), it has been noted that such models are rarely optimal, even if they were chosen by experts. For instance, in [15] the usefulness of classical distance-based measures for a classification task is being questioned for data sets with a high number of attributes. Additionally, a priori given distance-based similarity functions neglect a context for comparison of the objects.

Those observations were confirmed by psychologists studying properties of human perception of similar objects [41, 42, 51, 159]. One of the first researchers who investigated this problem was Amos Tversky. In 1977, influenced by results of his experiments on properties of similarity, he came up with *a contrast model* [159]. He argued that the distance-based approaches are not appropriate for modelling similarity relations due to constraints imposed by the mathematical features of the distance metrics such as the symmetry or subadditivity [159, 160]. Even the assumption about the representation in a multidimensional metric space was contradicted [13, 83, 160].

For instance, the lack of symmetry of a similarity relation is apparent when we consider examples of statements about similarity judgements such as "a son resembles his father" or "an ellipse is similar to a circle". Indeed, the experimental studies conducted by Tversky revealed that people tend to assign a significantly lower similarity scores when the comparison is made the other way around [159]. Moreover, even the reflexivity of the similarity is problematic, since in many situations a probability that an object will be judged by people as similar to itself is different for different objects [159].

In his model of a similarity Tversky proposed that the evaluation of a similarity degree was conducted as a result of a binary features matching process. In this approach, the objects are represented not as points in some metric space but as sets of their meaningful characteristics. Those characteristics should be qualitative rather than quantitative and their selection should take into consideration the context in which the similarity is judged. For example, when comparing cars in a context of their class (see discussion in Section 3.1.2) a relevant feature of a car could be that *its size is moderate* but a feature *its colour is red* probably does not need to be considered.

Tversky also noticed that the similarity between objects depends not only on their common features but also on the features that are considered distinct. Such features may be interpreted as arguments for or against the similarity. He proposed the following formula to evaluate the similarity degree of compared stimuli:

$$Sim_T(x, y) = \theta f(X \cap Y) - \Big(\alpha f(Y \setminus X) + \beta f(X \setminus Y)\Big), \tag{3.15}$$

where X and Y are sets of binary characteristics of the instances x, y, f is an interval scale function and the non-negative constants θ, α, β are the parameters. In Tversky's experiments f usually corresponded to the cardinality of a set.

Tversky argued that if the *ideal* similarity function for a given domain *sim* meets certain assumptions[5], there exist values of the parameters θ, α, β and an interval scale f that for any objects a, b, c, d, $Sim_T(a,b) > Sim_T(c,d) \Leftrightarrow sim(a,b) > sim(c,d)$.

Tversky's contrast model is sometimes expressed using a slightly different formula, known as the Tversky index:

$$Sim_T(x,y) = \frac{\theta f(X \cap Y)}{\theta f(X \cap Y) + \alpha f(Y \setminus X) + \beta f(X \setminus Y)}. \tag{3.16}$$

In this form values of the similarity function are bounded to the interval $[0,1]$. For appropriate values of the θ, α, β parameters and a selection of the interval scale function, this formula generalizes many common similarity functions. For example, if $\theta = \alpha = \beta = 1$ and f corresponds to the cardinality, Tversky index is equivalent to Jaccard similarity coefficient or Jaccard index[6]. When $\theta = 1$ and $\alpha = \beta = 0.5$ the Tversky's formula becomes equivalent to the Dice similarity coefficient.

Depending on the values of θ, α, β the contrast model may have different characteristics, e.g., for $\alpha \neq \beta$ the model is not symmetric. In Formula (3.15) $\theta f(X \cap Y)$ can be interpreted as corresponding to the strength of arguments for the similarity of x to y, whereas $\alpha f(Y \setminus X) + \beta f(X \setminus Y)$ may be regarded as a strength of arguments against the similarity. Using that model Tversky was able to create similarity rankings of simple objects, such as geometrical figures, which were more consistent with evaluations made by humans than the rankings constructed using standard distance-based similarity functions. Still, it needs to be noted that in those experiments, features to characterise the objects as well as the parameter settings were either chosen manually or they were extracted from results of a survey among volunteers who participated in the study. Although such an approach is suitable to explore small data, it would not be practical to use it for defining relevant features of objects described in large real-life data sets.

It is important to realize that in practical application, the features which can be used to characterize objects in the contrast model are usually on a much higher abstraction level than attributes from typical data sets. This fact makes it difficult to apply Tversky's model for predictive analysis of data represented in information systems. The problem is particularly evident when the analysed data are high dimensional. In such a case, manual construction of the important features is infeasible, even for domain experts.

For instance, microarray data sets contain numerical information about expression levels of tens of thousands genes. Within an information system, each gene corresponds to a different attribute. For such data, the appropriate features to use for

[5]Tversky made assumptions regarding viability of *the feature matching* approach, about *the monotonicity* of *sim* with regard to the common and distinct feature sets, *the independence* of the evaluation with regard to the common and distinct feature sets, *the solvability* of similarity equations and *the invariance* of the impact of particular feature sets on the similarity evaluation [159].

[6]It is easy to notice, that a function 1−Jaccard index corresponds to the binary distance discussed in Section 3.2.1.

Tversky's model may be interpreted as questions about activity of a particular gene or a group of genes, e.g.: *Are the Cytochrome C related genes overexpressed?* Since there is a huge number of genes and a function of many of them still remains unknown, experts are unable to manually select all the potentially important features of a given data sample. Additionally, there can be exponentially many binary characteristics for a data set and checking which of them can be used to characterize an object would be inefficient computationally. Those are the main motivations for a development of automated feature extraction methods and the similarity learning model which is proposed in Section 4.3.

3.2.3 Hierarchical and ontology-based similarity models

Similarity models are often built for very complex objects or processes with a predefined structure [6, 7, 10]. In such a case, a direct assessment of a similarity can be problematic, because two complex objects are likely to be similar in some aspects but dissimilar in other. Tversky's contrast model tries to overcome this issue by considering higher-level characteristics of objects and separately handling their common and distinctive features.

However, as it was pointed out in Section 3.2.2, typical data stored in information systems contain information only about relatively low-level, mostly numeric attributes. In order to define the higher-level features either domain knowledge or some learning techniques need to be employed. If the first eventuality is possible (i.e., an analyst has access to expert knowledge about the domain of interest), experts can provide description how to transform the attribute values into some more abstract but at the same time more informative characteristics.

For very complex objects a one aggregation step in construction of new features might be insufficient. Different features constructed from basic attributes might be correlated or might still require some generalization before they are able to express some relevant aspect of the similarity in a considered context. In this way, a whole hierarchy of features can be built. Such a structure is sometimes called *a similarity ontology* for a given domain.

Figure 3.3 shows a similarity ontology constructed for the car example discussed in previous sections. It was constructed for one of the data sets used as a benchmark in experiments described in Chapter 5 (i.e., the *Cars93* data). In this particular context (a class of a car) the similarity between two cars can be considered in aspects such as capacity, driving parameters, economy, size and value. Those local similarities can be aggregated to neatly express the global similarity, however the aggregation needs to be different for objects from different decision classes. For instance, the size aspect may be more important when assessing the similarity to a car from the *Full-size* class than in a case when the comparison is made to a *Sporty* car.

For this reason, in the hierarchical approach to approximating the similarity relation experts are required to provide local similarity functions and class-dependent aggregation rules. In this way the experts can give the model desirable properties. For example, even if only very simple distance-base local similarities are used for computation of the similarity in each of the aspects, the resulting model can still be not symmetric.

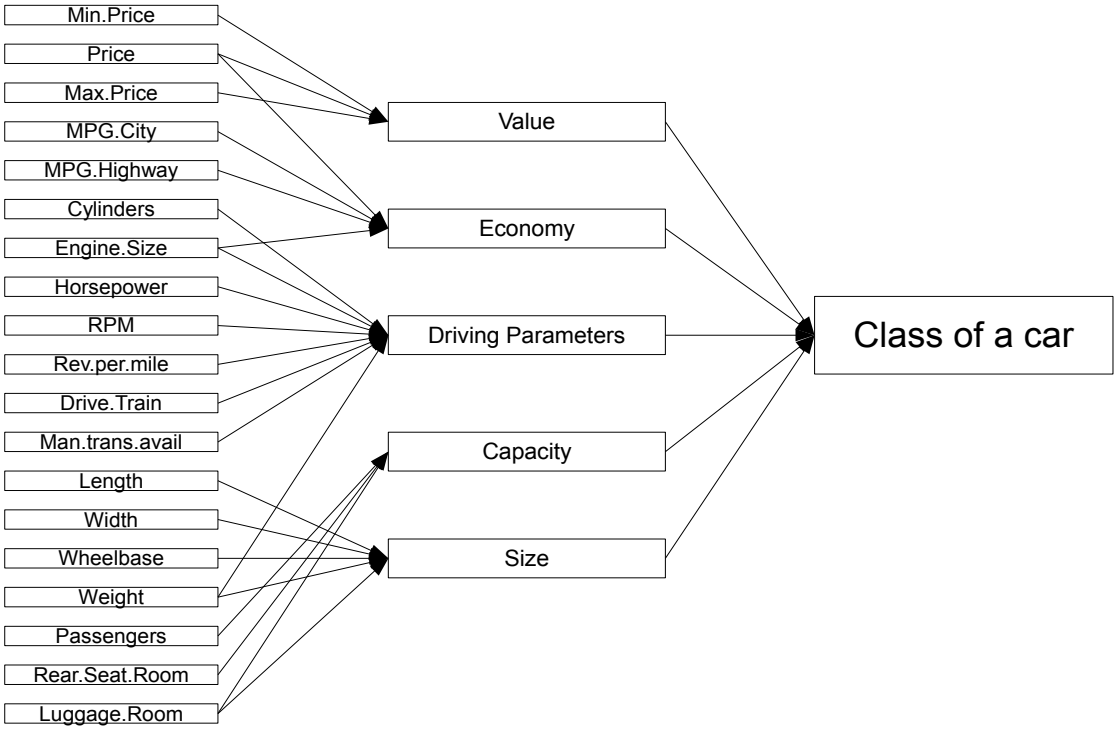


Figure 3.3: An exemplary similarity ontology for comparison of cars in the context of their type.

Some experiments with a hierarchical similarity models are described in [59, 60]. This approach was also successfully used for case-based prediction of a treatment plan for infants with respiratory failure [7, 10]. In that study, expert knowledge was combined with supervised learning techniques to assess the similarity of new cases in different aspects or abstraction levels. The incorporation of medical doctors into the model building process helped to handle the temporal aspect of data and made the results more intuitive for potential users.

One major drawback of the hierarchical similarity modelling is that it is extensively dependent on availability of domain knowledge. In the most of complex problems such knowledge is not easily obtainable. Additionally, the construction of a similarity ontology requires a significant effort from domain experts, which makes the model expensive. On the other hand, due to a vague and often abstract nature of the higher-level features which can influence human judgements of similarity, some expert guidance seems inevitable. Due to this fact, in practical applications the expert involvement needs to be balanced with automatic methods for learning the similarity from data.

3.3 Similarity in Machine Learning

Similarity models play an important role among the machine learning techniques. Their application ranges from supervised classification and regression problems to

automatic planning and an unsupervised cluster analysis. In this section, three major application areas of similarity models are discussed. They correspond to similarity-based classification models (Section 3.3.1), case-base reasoning framework (Section 3.3.2) and clustering algorithms (Section 3.3.3), respectively. Although the presented list of examples is by no means complete, it shows how useful in practice is the ability to reliably assess the similarity between objects.

3.3.1 Similarity in predictive data analysis and visualization

One of the most common application areas of the similarity modelling is the classification task. Models of similarity in this context can actually be constructed for two reasons. The first and obvious one is to facilitate classification of new, previously unseen objects, based on available data stored in an information system.

The most recognized similarity-based classification algorithm is the k-nearest neighbours [23, 89, 93, 107, 166]. It is an example of a lazy classification method which does not have a learning phase. Instead, for a given test case, it uses a predefined similarity measure to construct a ranking of the k most similar objects from a training data base (the neighbours). In the classical approach the measure is based on the Euclidean distance. The decision class of the tested object is chosen based on classes of the neighbours using some voting scheme [23, 53, 107]. This approach can be seen as an extension of the simplest similarity-based classification rule (Definition 3.1). It can be generalized even further by, for example, considering the exact similarity function values during the voting or assigning weights to training objects that express their representativeness for the decision class. The k-nearest neighbours algorithm can also be used to predict values of a numeric decision attribute (regression) or to perform a multi-label classification [68]. However, in all those applications the correct selection of a similarity model is the factor that has the biggest influence on the quality of predictions.

The models of a similarity in the classification context may also be constructed for a different purpose. The information about relations between objects from an investigated universe is sometimes as important as the ability to classify new cases. It can be used, for instance, to construct meaningful visualizations of various types of data [84, 153]. Such visualizations can be obtained by changing the representation of objects from original attributes to similarities. It allows to display the objects in a graph structure or a low-dimensional metric space. Such a technique is called *multidimensional scaling* (MDS) [13, 17].

Changing the representation of objects may also be regarded as a preprocessing step in a more complex data analysis process. For example, similarity degrees to some preselected cases can serve as new features. Such a feature extraction method (see [91]) can significantly improve classification results of common machine learning algorithms [22]. To make it possible, a proper selection of the reference objects is essential. One way of doing this requires a selection of a single object from each decision class, such that its average similarity to other objects from its class is the highest. Another possibility is the random embedding technique [164] in which the reference objects are chosen randomly and the quality of the selection is often verified on separate validation data.

3.3.2 Case-based Reasoning framework

The similarity-based classification can be discussed in a more general framework of algorithmic problem solving. Case-based reasoning is an example of a computational model which can be used to support complex decision making. It evolved from a model of dynamic memory proposed by Roger Schank [125] and is related to the prototype theory in cognitive science [119, 158].

A case-based reasoning model relies on an assumption that *similar problems*, also called cases, should have *similar solutions*. It is an analogy to the everyday human problem solving process. For example, students who prepare for a math exam usually solve exercises and learn proofs of important theorems, which helps them in solving new exercises during the test. The reasoning based on previous experience is also noticeable in work of skilled professionals. For instance, medical doctors diagnose a condition of a patient based on their experience with other patients with similar symptoms. When they propose a treatment, they need to be aware of any past cases in which such a therapy had an undesired effect.

In a typical case-based reasoning approach, each decision making or a problem solving process can be seen as a cycle consisting of four phases [1] (see Figure 3.4). In the first phase, called *retrieve*, a description of a new problem (case) is compared with descriptions stored in an available knowledge base and the matching cases are retrieved. In the second phase, called *reuse*, solutions (or decisions) associated with the retrieved cases are combined to create a solution for the new problem. Then, in the *revise* phase, the solution is confronted with the real-life and some feedback on its quality is gathered. Lastly, in the *retain* phase, a decision is made whether the new case together with the revised solution are worth to be remembered in the knowledge base. If so, the update is made and the new example extends the system.

The notion of similarity is crucial in every phase of the CBR cycle. The cases which are to be retrieved are selected based on their similarity degree to the new case. Often, it is required that those cases were not only highly similar to the reference object but that they were also maximally dissimilar to each other [1, 147]. In the reuse phase, the similarity degrees may be incorporated into the construction of the new solution, for example, as weights during a voting. Additionally, information about similarities between solutions associated with the selected cases may be taken into account during the construction of new ones. Next, during the revision of the proposed solution, its similarity to the truly optimal one needs to be measured, in order to assess an overall quality of the given CBR system and to find out what needs to be improved in the future. Finally, when the corrected solution to the tested case is ready, its similarity degrees to the cases from the knowledge base can be utilized again to decide whether to save the new case or not.

It is worth mentioning that the classical k-NN algorithm can be seen as a very basic CBR model [149], hence the similarity in a context of classification plays a special role in the case-based modelling. However, case-based reasoning may be used for solving much more complex problems than a simple classification, such as treatment planning or recognition of behavioural patterns [6, 7, 10]. The rough set theory has proven to be very useful for construction of CBR systems dedicated to complex problem solving [47, 57, 145].

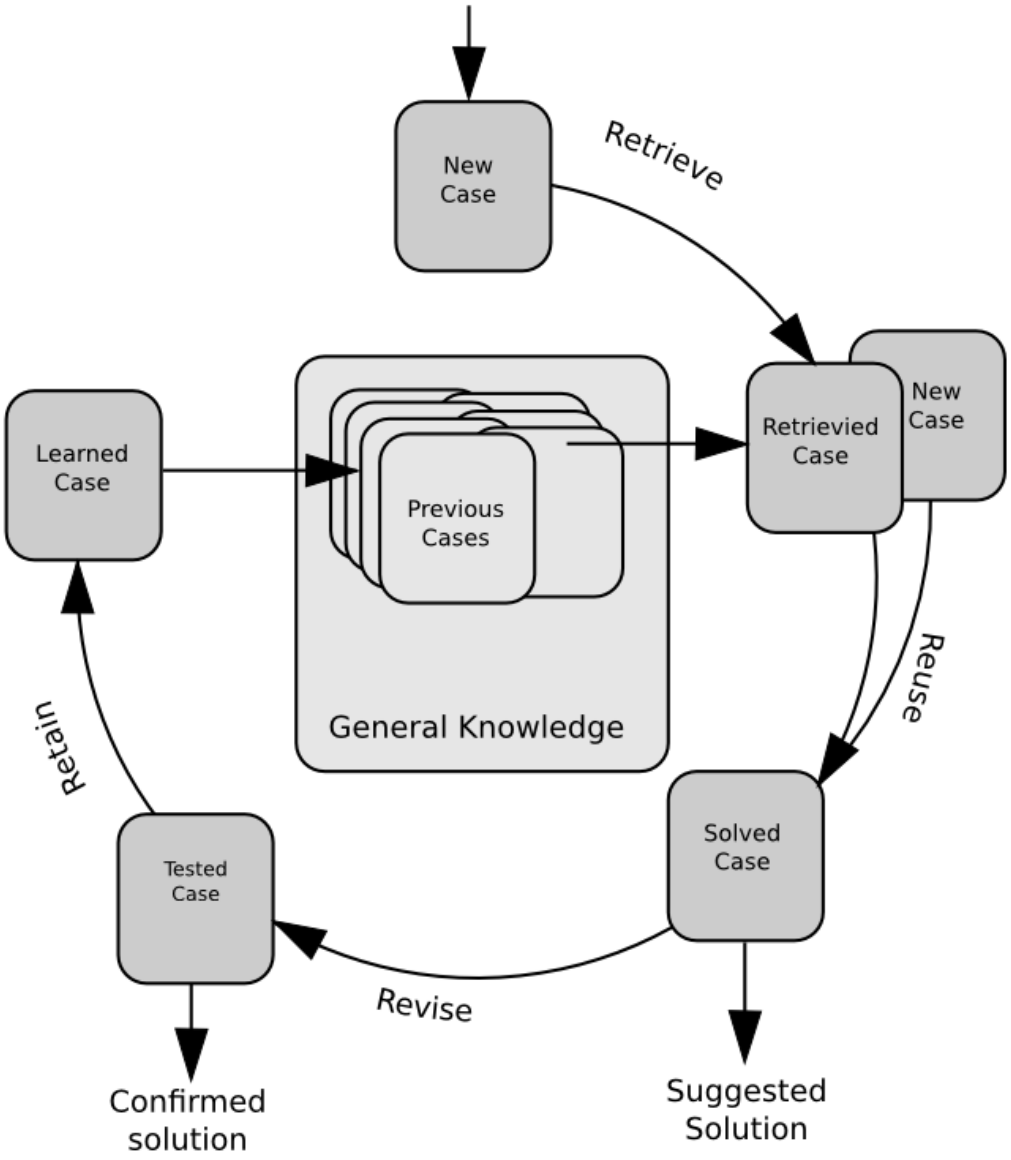


Figure 3.4: A full Case-Based Reasoning cycle (based on a schema from [1]).

3.3.3 Similarity in cluster analysis

The concept of similarity is also used for solving problems related with unsupervised learning. One example of such a task is clustering of objects into homogeneous groups [78, 93, 157].

In the clustering task the similarity can be used for two reasons. Since homogeneity of a cluster corresponds to the similarity between its members, similarity measures are used by clustering algorithms to partition objects into groups. The most representative example of such an algorithm is *k-means* [4, 78].

In the classical version of *k-means* objects are treated as points in the Euclidean space and the similarity between points is identified with their proximity. However, the algorithm can be easily modified to use any distance-based similarity function.

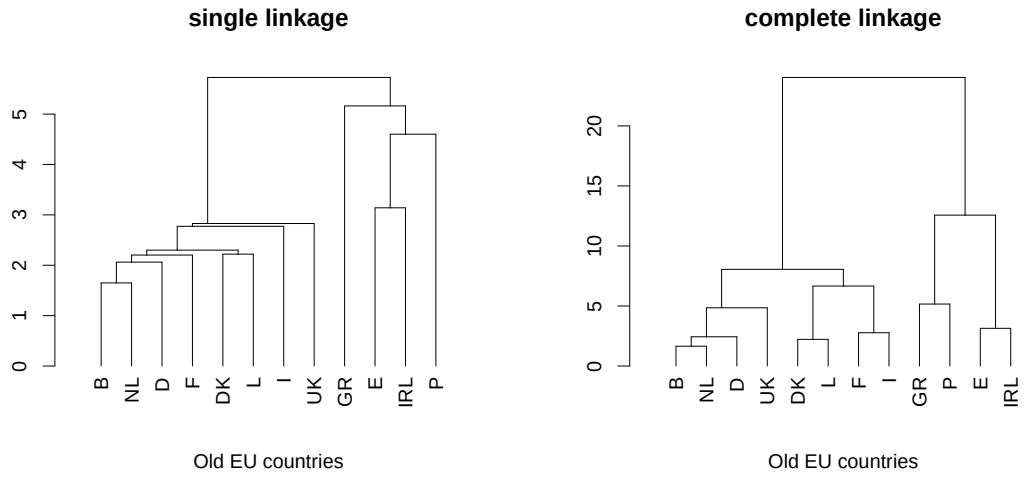


Figure 3.5: An example of two clustering trees computed for the *agriculture* data set using the *agnes* algorithm [78] with the *single* (on the left) and *complete* (on the right) linkage functions.

A pseudo code of such a modification of *k-means*, called *k-centroids*, is given below (Algorithm 1).

In typical implementations of *k-means*, when the similarity function is a linear function of Euclidean distance between points, the selection of new cluster centres is trivial. Coordinates of the new centres are equal to mean coordinates of the corresponding cluster members. However, if some non-standard similarity functions are used, the computation of the new centres requires solving an optimization problem and may become much more complex. Therefore, in many cases it is more convenient to use the *k-medoids* [78] algorithm which restricts the set of possible cluster centres to actual members of the group. This algorithm is also known to be more robust than *k-means* since it is not biased by outliers in the data [16, 78].

In the context of the clustering, it is also possible to consider a similarity between groups of objects (clusters). This notion is especially important for algorithms that construct a hierarchy of clusters. So-called *hierarchical clustering* methods, instead of dividing the objects into a fixed number of groups, compute a series of nested partitions with a number of groups ranging from 1 (all objects are in the same group) to the total number of objects in a data set (every object is a separate group). Figure 3.5 shows an example of two clustering trees computed for the *agriculture*[7] data set.

In the agglomerative approach to hierarchical clustering, at each iteration of the algorithm two most similar groups are merged into a larger one (the bottom-up approach). There can be many ways to estimate the similarity between two clusters. Typically, it is done using some *linkage function*. The most commonly used linkage functions are *single linkage*, *average linkage* and *complete linkage* [78, 157]. They estimate the similarity between two groups by, respectively, the maximum, average

[7]This data set describes a relation between a percentage of the population working in agriculture to Gross National Product (GNP) per capita in the old EU countries (in 1993).

Algorithm 1: The *k-centroids* algorithm.

Input: an information system $\mathbb{S} = (U, A)$;
a desired number of clusters k;
a similarity function $Sim : U \times U \to \mathbb{R}$;
Output: a grouping vector $g = (g_1, \ldots, g_{|U|})$, where $g_i \in \{1, \ldots, k\}$;

begin
  $endFlag = F$;
  Randomly select k initial cluster centres c_j, $j = 1, \ldots k$;
  while $endFlag == F$ **do**
    `Assign each` $u \in U$ `to the nearest (most similar) cluster centre:`
    for $i \in \{1, \ldots, |U|\}$ **do**
      $g_i = \arg\max_j \big(Sim(u_i, c_j)\big)$;
    end
    Compute new cluster centres $\bar{c}_j$, $j = 1, \ldots k$, such that $\sum_{i:g_i=j} \big(Sim(u_i, \bar{c}_j)\big)$ is minimal;
    if $\forall_{j \in \{1,\ldots,k\}} (\bar{c}_j == c_j)$ **then**
      $endFlag = T$;
    end
    else
      for $j \in \{1, \ldots, k\}$ **do**
        $c_j = \bar{c}_j$;
      end
    end
  end
  return $g = (g_1, \ldots, g_{|U|})$;
end

and minimum from similarities between pairs of objects, such that one object is in the first group and the other is in the second.

The second reason for using the similarity in the clustering task is related to the problem of evaluation of a clustering quality. This issue can be seen as a complement to the evaluation of similarity measures, which was discussed in Section 3.1.4. Given reference values of similarity degrees between pairs of considered objects (for instance by domain experts) it is possible to assess the semantic homogeneity of a grouping. It can be done, for example, by using a function that is normally employed as an *internal* clustering quality measure[8] but with the reference similarities as an input. Such an approach is utilized in experiments described in [70], as well as those presented in Section 5.3 of this dissertation.

[8] A clustering quality measure is called internal if its value is based solely on the data that were used by the clustering algorithm.

Chapter 4

Similarity Relation Learning Methods

The notion of similarity, discussed in the previous chapter, is a complex concept whose properties are subjective in nature and strongly depend on a context in which they are considered [13, 41, 83, 159, 160]. Due to this complexity, it is extremely difficult to model the similarity based only on an intuition and general knowledge about a domain of interest (see the discussion in Section 3.1). For decades this fact has motivated research on methods which would allow to approximate a similarity relation or to estimate values of a similarity function, using additional samples of data.

Many of the similarity learning methods concentrate on tuning parameters of some a priori given (e.g. by an expert) similarity functions. This approach is most noticeably present in the distance-based similarity modelling where the similarity function is monotonically dependent on a distance between representations of objects in an information system (for more details see Section 3.2.1). Distance measures can usually be constructed using the *local-global principle* [59, 102, 149] which divides the calculation of the distances into two phases – local, in which objects are compared separately on each of their attributes, and global, in which the results of comparisons are aggregated. This separation of the local and the global distance computation allows to conveniently parametrize the function with weights assigned to the local distances. Using available data and reference similarity values, those weights can be tuned in order to better fit the resulting similarity model to the given task.

Although the distance-based models for learning the similarity relation are predominant, they are not free from shortcomings. These defects are in a large part due to the usage of distance-based similarity function which can be inappropriate for modelling the similarity in a given context (see discussion in Sections 3.1.1, 3.1.2 and 3.2.2). Additionally, such an approach usually fails to capture higher-level characteristics of objects and their impact on the similarity relation. These limitations often lead to approximations of the similarity which are not consistent with human perception [42, 159].

To construct an approximation of the similarity which would truly mimic judgements of human beings it is necessary to go a step further than just relying on lower-level sensory data. The similarity learning process needs to support extraction of new higher-level characteristics of objects that might be important in the considered context. Since such abstract features are likely to correspond to vague concepts, some approximate reasoning methods need to be used in order to identify

their occurrence in the objects. Additionally, the aggregation of local similarities also needs to be dependent on data and should not enforce any specific algebraical properties on the approximated relation.

In this chapter, a flexible model for learning the similarity relation from data is proposed (in Section 4.3). This model, called Rule-Based Similarity (RBS), aims at overcoming the issues related with the distance-based approaches. As a foundation, it uses Tversky's feature contrast model (Section 3.2.2). However, unlike the feature contrast model, it utilizes the rough set theory to automatically extract higher-level features of objects which are relevant for the assessment of the similarity and to estimate their importance. In the RBS model the aggregation of the similarities in local aspects is based on available data and takes into consideration dependencies between individual features. The flexibility of this model allows to apply it in a wide range of domains, including those in which objects are characterised by a huge number of attributes.

In the subsequent sections some basic examples of similarity learning models are discussed. Section 4.1 explains the problem of similarity learning and points out desirable properties of a good similarity learning method. Section 4.2 is an overview of several approaches to similarity learning which mostly focus on tuning distance-based similarity functions. They utilize different techniques, such as attribute rankings, genetic algorithms or optimization heuristics, to select important attributes or to assign weights that express their relevance. On the other hand, Section 4.3 introduces the notion of Rule-Based Similarity whose focus is on constructing higher-level features of objects which are more suitable for expressing the similarity. Apart from explaining the motivation for the RBS model and its general construction scheme, some specialized modifications are presented. They adapt the model to tasks such as working with high dimensional data or learning a similarity function from textual data in an unsupervised manner.

4.1 Problem Statement

Similarity learning can be defined as a process of tuning a predefined similarity model or constructing a new one using available data. This task is often considered as a middle step in other data analysis assignments. If the main purpose for approximating a similarity relation is to better predict decision classes of new objects, facilitate planning of artificial agent actions or to divide a set of documents into semantically homogeneous groups, the resulting similarity model should help in obtaining better results than a typical baseline. Ideally, a process of learning the similarity should be characterised by a set features which indicate its practical usefulness.

The set of desirable similarity learning method properties include:

1. *Consistence with available data.*
   An ability to fit a similarity model to available data is the most fundamental feature of a similarity learning technique. It directly corresponds to an intuitive expectation that a trained model should be more likely to produce acceptable

similarity evaluations than an a priori given one. An outcome of a perfect method should always be a proper similarity function (see Definition 3.2), regardless of a data set. Moreover, this property should hold even for new objects that were not available for learning. Unfortunately, such a perfect method does not exist. A good similarity learning model, however, should aim to fulfil this intuition and be consistent with available data.

2. *Consistence with a context of the similarity that is appropriate for a given task.* A trained similarity model should also be consistent with a given context. Hence, if the context is imposed by, for example, a classification task, the resulting similarity model should be more useful for assigning decision classes of new objects using one of the similarity-based decision rules (see Definitions 3.1 and 3.2 in Section 3.1.3) than the baseline. The verification of the precision of such a classifier can be treated as a good similarity learning evaluation method [149]. This particular approach is used in the experiments described in the next chapter.

3. *Ability to take into consideration an influence of objects from the data on evaluation of the similarity.*
As it was mentioned in Section 3.1.2, similarity between two given objects often depends on the presence of other objects which are considered as a kind of a reference for comparison. Similarity learning methods that are able to construct a similarity model capable of capturing such a dependence are justifiable from the psychological point of view and are more likely to produce intuitive results [42, 51].

4. *Compliance with psychological intuitions (e.g. regarding object representations).* Another desirable property of similarity learning models is also related to intuitiveness of the resulting similarity evaluations. Assessments of similarity obtained using constructed similarity function should be comprehensible for domain experts. One way of ensuring this is to express the similarity in terms of meaningful higher-level features of objects. Such features can be extracted from data using standard feature extraction methods [91] as well as with specialized methods such as decision rules [62, 65, 128] or semantic text indexing tools [38, 155]. Not only can higher-level features help in capturing aspects of similarity that are difficult to grasp from lower-level sensory data but may also be used as a basis for a set representation of objects [51, 159]. Such representation can be more natural for objects that are difficult to represent in a metric space [41, 42, 159]. Moreover, by working with higher-level features the similarity evaluation can be associated with resolving conflicts between arguments for and against the similarity of given objects. Such an approach is usually more intuitive for human experts.

5. *Robustness for complex object domains (e.g. high dimensional data).*
A good similarity learning method should be general enough to be possible to apply in many object domains. Usually, a similarity model is efficient for some data types while for others it yields unreliable results. The usage of a similarity learning technique for tuning parameters of a model can greatly extend the

range of suitable data types. However, applications of a similarity learning method may also be confined. For instance, models with multiple parameters are more vulnerable to overfitting when there is a limited number of available instances in data (e.g. the few-objects-many-attributes problem [15, 139]).

6. *Computational feasibility.*
The last of the considered properties regards computational complexity of the similarity learning model. The complexity of a model can be considered in several aspects. A similarity learning method needs to be computationally feasible with regard to the size of a training data set, understood in terms of both, the number of available objects and the number of attributes. Either of those two sides can be more important in specific situations. Many models, however, are efficient in one of the aspects and inefficient in the other. The scalability of a similarity learning model often determines its practical usefulness.

Any similarity learning model can be evaluated with regard to the above characteristics. In particular, Rule-Based Similarity described in Section 4.3 was designed to possess all those properties.

4.2 Examples of Similarity Learning Models

The problem of similarity learning was investigated by many researchers from the field of data analysis [21, 28, 47, 61, 74, 90, 102, 149, 170]. In the applications discussed in Section 3.3, similarity functions which can be employed for a particular task can be adjusted to better fit the considered problem. The main aim of such an adjustment is to improve effectiveness of the algorithms which make use of the notion of similarity.

The commonly used similarity models (e.g. the distance-based models – see Section 3.2.1) neglect the context for similarity. However, the vast majority of similarity learning methods incorporate this context into the resulting model by, e.g., considering feedback from experts or by guiding the learning process using evaluations of the quality of the model computed on training data. Thus, in a typical case, the similarity learning can be regarded as a way of adaptation of a predefined similarity model to the context which is determined by a given task.

The process of learning the similarity relation may sometimes be seen as a supervised learning task. This is especially true when it can be described as a procedure in which an omniscient oracle is queried about similarities between selected pairs of objects to construct a decision system for an arbitrary classification algorithm [47, 90]. However, in many cases, direct assessments of the degrees of similarity which can be used as a reference are not available. In that situation, domain knowledge or some more general properties of the similarity in the considered context have to be used to guide the construction of the model. One example of such a property is stated in Definition 3.1. Since the later approach can be regarded as more practical, the following examples show similarity learning methods mainly designed to work in such a setting.

4.2.1 Feature extraction and attribute ranking methods

One of the most general methods for learning a similarity relation is to adjust the corresponding similarity function to a given data set by assigning weights, selecting relevant attributes or constructing new, more informative ones. Such weights can be used in combination with standard generalizations of similarity functions (e.g. a measure based on the generalized Minkowsky distance – see Section 3.2.1) to express the importance of particular local similarities.

Research on attribute selection techniques has always been in a scope of interest of the machine learning and data mining communities [94, 99, 100, 162]. The dimensionality reduction allows to decrease the amount of computational resources needed for execution of complex analysis and very often leads to better quality of the final results [50, 74, 91]. The selection of a small number of meaningful features also enables better visualizations and can be crucial for human experts who want to gain insight into the data.

Feature selection methods can be categorised in several ways. One of those is the distinction between supervised and unsupervised algorithms. The unsupervised methods focus on measuring variability and internal dependencies of attributes in data. As an example of such a method one can give Principle Component Analysis [76] in which the representation of data is changed from original attributes to their representation in a space of eigenvectors, computed by eigenvalue decomposition of an attribute correlation matrix. The supervised methods information about decision classes to assess the relevance of particular attributes. They can be further divided into three categories, i.e. filter, wrapper and embedded methods [75, 81, 91].

The filter methods create rankings of individual features or feature subsets based on some predefined scoring function. Ranking algorithms can be divided into univariate and multivariate. The univariate rankers evaluate importance of individual attributes without taking into consideration dependencies between them. A rationale behind this approach is that a quality of an attribute should be related to its ability to discern objects from different decision classes. As an example of frequently used univariate algorithms one can give a simple correlation-based ranker [52], statistical tests [87] or rankers based on mutual information measure [116]. The multivariate attribute rankers try to assess the relevance in a context of other features. They explore dependencies among features by testing their usefulness in groups (e.g. the relief algorithm [79]) or by explicitly measuring relateness of pairs of attributes and applying the minimum-redundancy-maximum-relevance framework [31, 116]. Another worth-noticing example of a multivariate feature ranker is the Breiman's relevance measure. It expresses the average increase of a classification error resulting from randomization of attributes that were used during construction of trees by the Random Forest algorithm [20, 32].

In the second approach, subsets of features are ranked based on the performance of a predictive model constructed using those features. Attributes from the subset which achieved the highest score are selected. Usually, the same model is used for choosing the best feature set and making predictions for test data, because different classifiers may produce their best results using different features. Due to the fact that a number of all possible subsets of attributes is exponentially large, different heuristics are being used to search the attribute space. The most common heuristics

include top-down search [88], bottom-up search [165] and Monte Carlo heuristics such as simulated annealing or genetic algorithms [129]. Although usually the wrapper approach yields better results than the filter approach, its computational complexity makes it difficult to apply for extremely high dimensional data.

Table 4.1: Summary of attribute selection methods.

Filter methods:	Wrapper methods:	Embedded methods:
• attributes or attribute subsets receive scores based on some predefined statistic, • scores of individual attributes can be used as weights, • top ranked features can be selected as relevant.	• learning algorithms are evaluated on subsets of attributes, • many different subset generation techniques can be used, • the best subset is selected.	• feature selection can be an integral part of a learning algorithm, • irrelevant attributes may be neglected, • some new features may be constructed (internal feature extraction).

The embedded methods are integral parts of some learning algorithm. For instance, classifiers such as Support Vector Machine (SVM) [37, 163] can work in a space of higher dimensionality than the original data applying the kernel trick [126]. Moreover, efficient implementations of classifiers such as Artificial Neural Networks (ANN) [44, 167] automatically drop dimensions from the data representation if their impact on the solution falls below a predefined threshold.

The application of an attribute selection or ranking algorithm for learning a similarity may be dependent on its context. Practically all typical supervised feature selection algorithms can be employed for tuning a similarity function if the similarity is considered in the classification context. It is a consequence of the main feature of similarity in that context (see Definition 3.1). If the similarity needs to be consistent with decision classes, the more discriminative attributes are likely to be relevant in a similarity judgement. However, in the case of similarity in the context of "general appearance" unsupervised feature extraction methods need to be used.

4.2.2 Genetic approaches

Genetic algorithms (GA) [92] are another popular tool for learning the parameters of similarity functions which are constructed using the *local-global principle*. The idea of GA was inspired by evolution process of living beings. In this approach parameters of the local similarities (e.g. their weights) and the aggregation function are treated as genes and are arranged into genotypes of the genome (also called

chromosomes). In this nomenclature, the similarity learning process corresponds to searching for the most adapted genotype. The adaptation of a genotype to a given problem is measured using a fitness function. Since in applications to similarity learning a proper fitness function needs to be based on a similarity model evaluation method, such as those discussed in Section 3.1.4, the GA-based similarity learning may be regarded as a special case of the wrapper attribute ranking approach (see Section 4.2.1). However, the flexibility of GA makes it particularly popular among researchers from the case-base reasoning field [28, 74, 149].

In GA searching for the most adapted genotype is iterative. In each iteration, which is also called a life-cycle or a generation, chromosomes undergo four genetic operations, namely the replication (inheritance), mutation, crossover and elimination (selection).

1. Replication – a selected portion of genotypes survives the cycle and is carried out to the next one.

2. Mutation – a part of genotypes is carried out to the next generation with a randomly modified subset of genes.

3. Cross-over – some portion of genotypes exchange a part of their genetic code and generate new genotypes.

4. Elimination – a part of genotypes that were taken to the new population is removed based on their values of a fitness function.

Figure 4.1 presents a schema of an exemplary genetic optimization process (a genetic life-cycle). Initially, a random population of genotypes is generated, with each genotype coding a set of parameters of a similarity function that is being tuned. The genetic operations are repeatedly applied to consecutive populations until stop criteria of the algorithm are met.

Exact algorithms for performing the genetic operations may vary in different implementations of GA. However typically, the selection of genotypes to undergo the replication, mutation and crossover is non-deterministic. It usually depends on scores assigned by the fitness function. A common technique for selecting genotypes is called the roulette wheel selection – every member of a population receives a certain probability of being selected and the genotypes are chosen randomly from the resulting probability distribution. The selection of genotypes for different genetic operations is done independently, which means that a single genotype may undergo a few different genetic operations. It can also be chosen several times for the same operation type.

During the replication, selected genotypes are copied unchanged to the next generation. The mutation usually involves random selection of a relatively small subset of genes, which are then slightly modified and the resulting genotype is taken to the next cycle. The crossover operation is usually the most complex. Its simplest exemplary implementation may consist of swapping randomly selected genes between two genotypes. If all parameters of a similarity function are numeric, it may also be realized by computing two weighted averages of the parent genotypes. One which gives more weight to genes from one parent and the second giving a higher weight

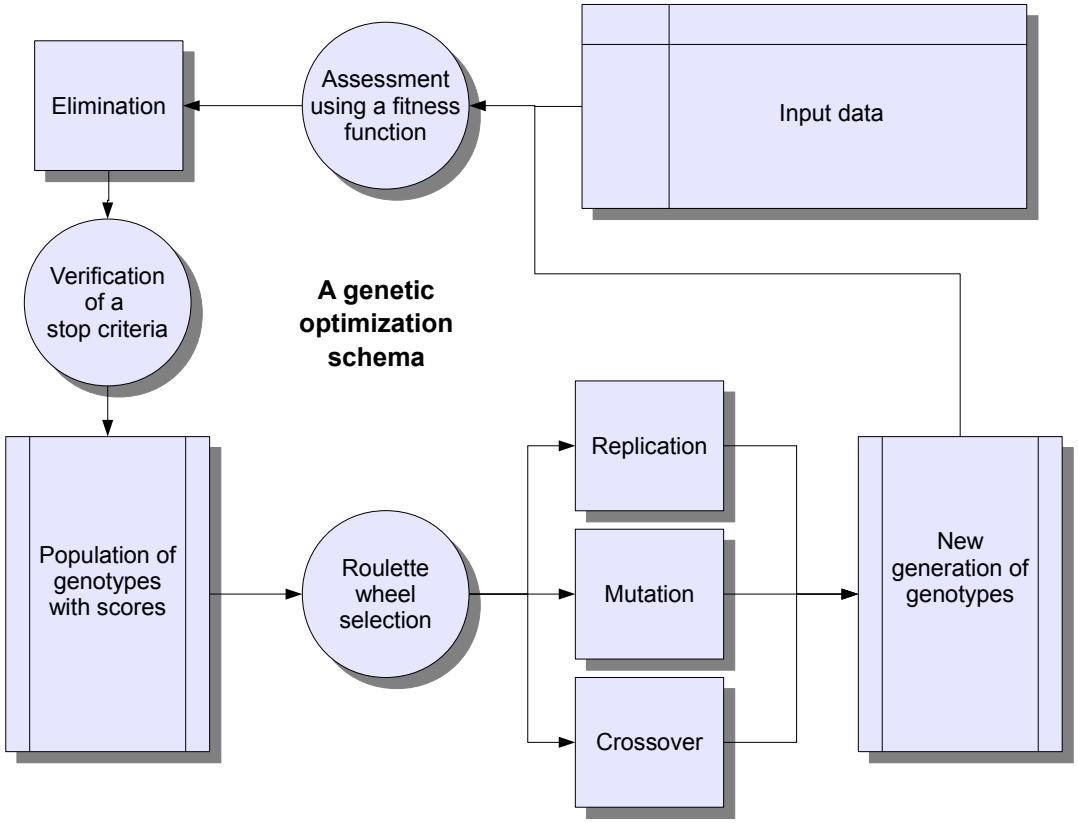


Figure 4.1: A schema of a single genetic life-cycle.

to the other one. A more complicated variants of the crossover may include the construction of completely new features that are used for measuring the similarity or to define new local similarity measures [149]. The elimination of genotypes is performed to maintain a desired size of a population. Most commonly, genotypes with the lowest values of a fitness function are removed before starting a new life-cycle. In some implementations however, this last phase of the cycle is also done in a non-deterministic manner, using techniques such as the roulette wheel selection.

The most computationally intensive part of GA is the quality evaluation of genotypes belonging to the population. In the context of the similarity learning, the fitness function needs to assess a quality of a similarity function with parameters corresponding to each genotype in the population. Those assessments are usually performed using one of the methods discussed in Section 3.1.4 and they require a comparison of similarity values returned by the tested models on a training data set with some reference.

4.2.3 Relational patterns learning

A different model for learning the similarity relation from data was proposed among the relational patterns learning methods [102]. This approach also employs the *local-global principle* for defining approximations of the similarity. However, it differs from the previously discussed methods in that it tries to directly approximate the

Table 4.2: An exemplary data set describing a content of two vitamins in apples and pears (the data were taken from [102]).

Vitamin A	Vitamin C	Fruit	Vitamin A	Vitamin C	Fruit
1.0	0.6	Apple	2.0	0.7	Pear
1.75	0.4	Apple	2.0	1.1	Pear
1.3	0.1	Apple	1.9	0.95	Pear
0.8	0.2	Apple	2.0	0.95	Pear
1.1	0.7	Apple	2.3	1.2	Pear
1.3	0.6	Apple	2.5	1.15	Pear
0.9	0.5	Apple	2.7	1.0	Pear
1.6	0.6	Apple	2.9	1.1	Pear
1.4	0.15	Apple	2.8	0.9	Pear
1.0	0.1	Apple	3.0	1.05	Pear

similarity in a local distance vector space. The learning in the context of classification is done through optimizing a set of parameters for an a priori given family of similarity approximations. Since the usage of distance-based local similarities enforces reflexivity and symmetry on the resulting approximation, this approach is highly related to searching for optimal approximations in tolerance approximation spaces [120, 133, 134].

The first step in relational patterns learning algorithms is a transformation of data from an attribute value vector space into a local distance (or similarity) vector space. This process for an exemplary *fruit* data set (Table 4.2) taken from [102] is depicted on Figure 4.2.

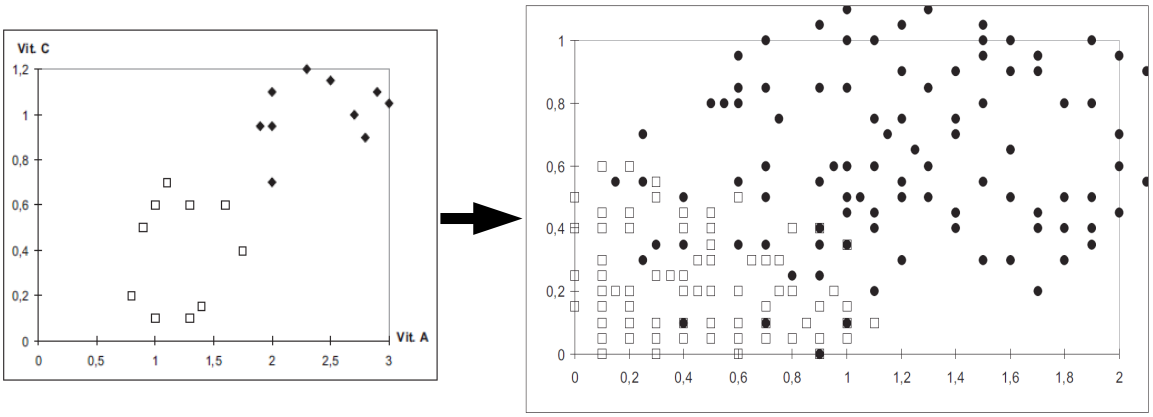


Figure 4.2: An example of a transformation of the fruit data set (Table 4.2) from the attribute value vector space to the local distance vector space. In the plot on the left, the white squares correspond to apples and the black diamonds represent pears. In the plot on the right, the squares correspond to pairs of instances representing the same fruit, whereas the black circles are pairs of different fruits (an apple and a pear).

For each pair of objects from a decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$, all their local distances (or sometimes local similarities) are computed. Those values are used to represent the pairs in a new metric space, whose dimensionality is the same as the total number of original attributes. A new binary decision attribute is also constructed. It indicates whether the both of objects from the corresponding pair belong to the same decision class of original data. This new data representation can serve as an approximation space (see Section 2.2.3) for learning the similarity in the context of classification (see Definition 3.1).

The selection of the most suitable approximation from a given family is performed by searching for parameters that maximize the number of pairs with the same decision (white squares in the right plot of Figure 4.2) included in the approximation while maintaining the constraints resulting from Definition 3.1. One example of a family of approximations is the *parametrized conjunction* in a form:

$$(x, y) \in \tau_1^*(\epsilon_1, \ldots, \epsilon_{|A|}) \Leftrightarrow \bigwedge_{a_i \in A} \Big[f_{a_i}(x, y) < \epsilon_i \Big], \tag{4.1}$$

where A is a set of all conditional attributes, f_{a_i} are local distance functions and ϵ_i are parameters of the family. A different example of a useful family of approximations the *parametrized linear combination* form:

$$(x, y) \in \tau_2^*(w_0, \ldots, w_{|A|}) \Leftrightarrow \sum_{i=1}^{|A|} \Big[w_i \cdot f_{a_i}(x, y) \Big] + w_0 < 0, \tag{4.2}$$

where w_i, $i = 0, \ldots, |A|$ are parameters. Those two families of approximations differ in geometrical interpretations of neighbourhoods they assign to the investigated objects. In the first one, the captured similar objects need to be in a rectangular-shaped area, whereas in the case of the second family, the neighbourhoods are diamond-like. Several heuristics for learning semi-optimal sets of parameters for different families of similarity relation approximations are shown in [102].

One disadvantage of the relational pattern learning approach is its computational complexity. The transformation of the original decision system into a local distance vector space requires $O(n^2)$ storage and computation time (n is the number of objects in the decision system). In order to avoid such a big computational cost, the transformation may be virtualized, i.e. it may not be physically performed but instead the computation of local distances and new decision value may be done "on demand" by the learning heuristic. Although such a solution decreases the space complexity, it usually leads to a significant increase in the time complexity of the method.

Another solution is to approximate similarity only to a selected small subset of objects from the decision system. This technique, called *local approximation* of the similarity, might be efficient, especially when it is possible to distinguish a small group of objects which are representative for the whole data.

4.2.4 Explicit Semantic Analysis

Many similarity learning models were proposed specifically to approximate a semantic similarity in corpora of textual data [25, 35, 38, 55]. One of the most successful

approaches aims at improving a representation of documents, so that it better reflected a true meaning of the texts [25, 38]. With the new representation, the similarity of two documents is estimated using standard similarity measures, such as those described in Section 3.2.1.

One particularly interesting example of such a method is Explicit Semantic Analysis (ESA), proposed in [38]. It is based on an assumption that any document can be represented by predefined concepts which are related to the information that it carries (its semantics). Those concepts can be then treated as semantic features of documents. The process of choosing concepts that describe documents in the most meaningful way can be regarded as a feature extraction task [91].

In the ESA approach, natural language definitions of concepts from an external knowledge base, such as an encyclopaedia or an ontology, are matched against documents to find the best associations. A scope of the knowledge base may be general (like in the case of Wikipedia) or it may be focused on a domain related to the investigated text corpus, e.g. Medical Subject Headings (MeSH)[1] [161]. The knowledge base may contain some additional information on relations between concepts, which can be utilized during computation of the "concept-document" association indicators. Otherwise, it is regarded as a regular collection of texts with each concept definition treated as a separate document.

The associations between concepts from a knowledge base and documents from a corpus are treated as indicators of their relatedness. They are computed two-fold. First, after the initial preprocessing (stemming, stop words removal, identification of relevant terms), the corpus and the concept definitions are converted into the *bag-of-words* representation. Each of the unique terms in the texts is given a set of weights which express its association strength to different concepts.

Assume that after the initial processing of a corpus consisting of M documents, $D = \{T_1, \ldots, T_M\}$, there have been identified N unique terms (e.g. words, stems, N-grams) $w_1, \ldots, w_N$. Any text T_i in the corpus D can be represented by a vector $\langle v_1, \ldots, v_N \rangle \in \mathbb{R}_+^N$, where each coordinate $v_j(T_i)$ expresses a value of some relatedness measure for j-th term in the vocabulary (w_j) relative to this document. The most common measure for calculating $v_j(T_i)$ is the *tf-idf* (term frequency-inverse document frequency) index (see [35]) defined as:

$$v_j(T_i) = tf_{i,j} \cdot idf_j = \frac{n_{i,j}}{\sum_{k=1}^{N} n_{i,k}} \cdot \log\left(\frac{M}{|\{i : n_{i,j} \neq 0\}|}\right), \tag{4.3}$$

where $n_{i,j}$ is the number of occurrences of the term w_j in the document T_i.

In the second step, the bag-of-words representation of concept definitions is transformed to an inverted index which maps words into lists of K concepts, $c_1, \ldots, c_K$, described in a knowledge base. The inverted index is then used to perform a semantic interpretation of documents from the corpus. For each text, the semantic interpreter iterates over words that it contains, retrieves corresponding entries from the inverted index and merges them into a vector of concept weights (association strengths) that represent a given text.

[1] MeSH is a controlled vocabulary and thesaurus created and maintained by the United States National Library of Medicine. It is used to facilitate searching in life sciences related article databases.

Let $W_i = \langle v_j \rangle_{j=1}^{N}$ be a bag-of-words representation of an input text T, where v_j is a numerical weight of a word w_j expressing its association to the text T_i (e.g. its tf-idf). Let $inv_{j,k}$ be an inverted index entry for w_j, where $inv_{j,k}$ quantifies the strength of association of the term w_j with a knowledge base concept c_k, $k \in \{1, \ldots, K\}$. The new vector representation of T_i, called a *bag-of-concepts*, will be denoted by $C_i = \big(c_1(T_i), \ldots, c_K(T_i)\big)$, where

$$c_k(T_i) = \sum_{j:w_j \in T_i} v_j \cdot inv_{j,k} \tag{4.4}$$

is a numerical association strength of k-th concept to the document T_i. In Section 5.3, texts will also be represented as a set of concepts with sufficiently high association level denoted by $F_i = \{f_k : c_k(T_i) \geq minAssoc_k\}$. Those concepts will be treated as binary semantic features of texts, such as those which are utilized by Tversky's contrast model [159] (see Section 3.2.2).

The representation of texts by sets of features can be easily transformed into an information system $\mathbb{S} = (D, F)$, where $F = \bigcup_{i=1}^{|D|} F_i$. Each possible feature of documents is treated as a binary attribute in $\mathbb{S}$. The semantic similarity between objects from $\mathbb{S}$ can be assessed using standard measures described in Section 3.2. However, due to sparsity and high dimensionality of this representation, usually spherical similarity functions, such as the *cosine similarity*, or set-oriented measures as *Jaccard index* and *Dice coefficient*, are employed. In several papers it is experimentally shown that this representation can yield better evaluations of the similarity than the standard bag-of-words [38, 70, 155].

If the utilized knowledge base contains additional information on semantic dependencies between the concepts, this knowledge can be used to further adjust the vector (4.4). Moreover, if experts could provide feedback in the form of manually labelled exemplary documents, some supervised learning techniques can also be employed in that task [72]. However, particular methods for automatic tagging of textual data are not in the scope of this research.

4.3 Rule-Based Similarity Model

This section presents a similarity learning model which is the main contribution of this dissertation. The model, called Rule-Based Similarity (RBS), originally proposed in [62] and reared in [61, 64, 65, 67, 70], was inspired by works of Amos Tversky. It can even be seen as a rough set extension of the psychologically plausible feature contrast model proposed in [159] (see also the discussion in Section 3.2.2).

Tversky's model is extended in a few directions. In RBS, some basic concepts from the rough set theory are used to automatically extract from available data, features that influence the judgement of similarity. Additionally, the proposed method for aggregation of local similarities and dissimilarities takes into consideration dependencies between the induced features that occur in data. This allows for a more reliable assessment of the importance of arguments for and against the similarity of investigated objects. Finally, the simplicity and flexibility of RBS makes it useful in a wide array of applications, including learning the similarity in a classification context

from both regular and extremely high dimensional data. It can also be modified to allow learning the semantic similarity of textual documents.

The following sections overview the construction of RBS in different application scenarios. Section 4.3.1 explains the main motivation behind the model and points out its relations to Tversky's feature contrast model. Next, Section 4.3.2 shows how the basic RBS model is constructed and then, Section 4.3.4 discusses an adaptation of RBS to the case when data describing considered objects are high dimensional. The last section (Section 4.3.5) shows how RBS can be adjusted to work in an unsupervised fashion, especially for learning a similarity measure appropriate for assessment of the similarity in a meaning of texts.

4.3.1 General motivation for Rule-Based Similarity

The similarity learning models discussed in Section 4.2 allow to fit a parametrized similarity function or a family of approximation formulas to available data. This process can be understood as an adjustment of a similarity relation to a desired context. However, in case of the discussed methods this problem is reduced to tuning parameters of a preselected similarity model. An approach like that has to result in passing to the final model some properties which are not inferred from data and are potentially unwanted.

The approach to learning the similarity represented by the commonly used similarity models is usually based on an assumption that an expert is able to preselect at least a proper family of similarity models. This family is expected to contain a member which can sufficiently approximate the reality. Unfortunately, due to the complexity of the concept of similarity, this assumption may be false. Additionally, in some cases the family of possible approximations may be so large, that the extensive parameter tuning is likely to terminate at some relatively weak local optimum or even to overfit to training data while showing poor performance when used for new, previously unknown objects.

This problem is particularly conspicuous when the analysed data set is high dimensional. Typically, the number of parameters of a similarity learning model is at least linearly dependent on the number of attributes in data. Hence, dimensionality has a significantly adverse impact on a complexity of a model. Not only can very complex models suffer from overfitting but they are also unintuitive and difficult to interpret by experts.

Another important issue related with the similarity learning methods which utilize the *local-global principle* is a difficulty with modelling dependencies between local similarities/distances corresponding to different attributes. For instance, what weights should be assigned to a group of highly correlated attributes which are important for the similarity judgement individually? On one hand, each local similarity is important so it should have a high weight. On the other hand, if all of those local similarities are given a high weight, the final model can be biased towards a single aspect of a similarity while neglecting other, possibly as relevant factors.

Moreover, in comparison to approximation of concepts, approximation of relations often requires extraction of some additional higher-level characteristics related to pairs of objects (see the discussion in Sections 2.2 and 3.1). Figure 4.3 shows a general

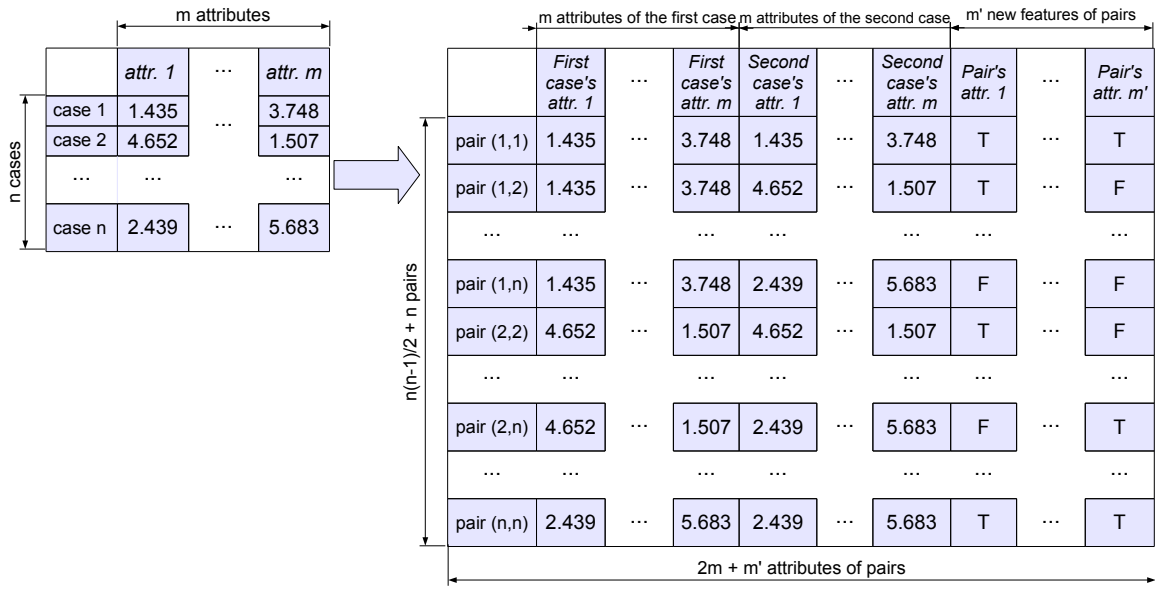


Figure 4.3: A schema showing generalized tabular representations of an information system for the purpose of learning a similarity relation. An information system on the left is transformed into a relational pattern learning space represented on the right.

transformation schema of an information system that allows flexible learning of any similarity relation. Apart from attributes that describe each object, the transformed system may contain additional features that characterise two objects as a pair. Such features may correspond to a variety of statistics or different aggregation types of values of the original attributes. The models discussed in Section 4.2 simplifies this transformation by narrowing the set of new characteristics to predefined local similarities. Even though such a limitation is beneficial from the computational complexity point of view, it may severely deteriorate the model's ability to infer a semantically meaningful similarity relation. It was confirmed in a number of empirical studies that the introduction of higher-level features often significantly increases performance of similarity models [7, 21, 38, 59, 60].

The need for extraction of meaningful qualitative features of objects for the purpose of measuring their similarity was recognized by Tversky and motivated his contrast model [159] (see Section 3.2.2). Tversky argued that people rarely think in terms of exact numbers but instead they tend to operate on binary characteristics of objects, such as *an object is large* or *an object is round*. In his model, objects were represented by such higher-level features. For each pair, the features were divided into those which are arguments for the similarity (the common features) and those which constitute arguments against the similarity or, in other words, arguments for dissimilarity of the objects from the pair. The RBS model is in a large part inspired by this approach.

In Tversky's experiments relevant characteristics of the compared stimuli were usually defined by participants of the conducted study. However, for analysis of larger real-life data sets, meaningful features need to be extracted automatically. Their selection and influence on the final model needs to be dependent on a context of the

similarity relation and, in particular, on other objects from the given data set. One of the main aims of the RBS model is to facilitate this task using a rough-set-motivated approach for approximation of relations (see Section 2.2.3).

Semantically meaningful higher-level features of objects can be extracted from data using a rule mining algorithm [62, 128]. Unlike in [128], however, in RBS such features are not only used for changing the representation of objects but are also utilized to construct approximations of similarities to each object in a training data set. A RBS similarity function value is derived from those approximations to allow convenient modelling of the dependencies imposed by the presence of different objects in the data (see the discussion in Section 3.1.2).

Another goal of RBS is to overcome the problem with selection of appropriate weights for the contrast model. Instead of assigning globally defined importance values to common and distinctive features of any pair, RBS aims at assessing strength of all arguments for and all arguments against the similarity, relative to an investigated pair. This approach allows RBS to better reflect the context in which the similarity of given objects is considered.

Finally, a good similarity learning model need to be scalable. The scalability of a model can be considered relative to a number of objects in the data as well as to a number of attributes. Construction of RBS does not require investigating all pairs of object during the learning, hence it is possible to approximate a similarity relation from larger data. By utilizing basic notions from the theory of rough sets, RBS can also be adapted to work with extremely high dimensional data.

In general, construction of a desired similarity learning model should include the following steps:

1. The selection of an appropriate context for the similarity.
2. The extraction of features which are relevant in the given context (definition of an approximation space).
3. The definition of a data-dependent similarity function that aggregates the features, while considering the preselected context and types of compared objects.

The next section shows how the RBS model implements these steps in order to incorporate the properties discussed in Section 4.1.

4.3.2 Construction of the Rule-Based Similarity model

The Rules-based Similarity (RBS) model was developed as an alternative to the distance-based approaches [62]. It may be seen as a rough set extension to the psychologically plausible feature contrast model proposed by Tversky [159]. As in the case of the contrast model, in RBS the similarity is assessed by examining whether two objects share some binary higher-level features. Unlike in Tversky's approach, however, in RBS features that are relevant for a considered similarity context are automatically extracted from data. Their importance is also assessed based on available data, which allows to model the influence of information about other objects on the similarity judgement (see the discussion in Section 3.1).

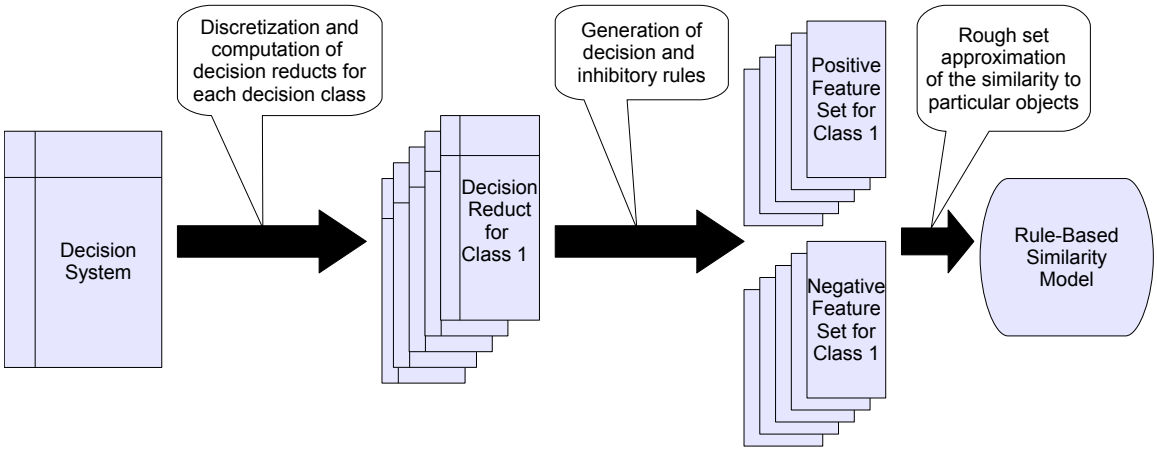


Figure 4.4: A construction schema of the RBS model.

The construction of RBS is performed in three steps. The schema from Figure 4.4 shows those steps in a case when the similarity is learnt in a classification context (the context is defined by a decision attribute in a data set). Since originally the notion of the RBS model was proposed for a classification purpose, that specific context will be assumed for the remaining part of this section.

The first step involves transformation of raw attribute values, which often are numerical, into a more abstract symbolic representation that resembles basic qualitative characteristics of objects. As discussed in Sections 3.1 and 3.2.2, such characteristics are more likely to be used by humans and are more suitable for an assessment of a local similarity from a psychological point of view [42, 51, 119, 158, 159, 160]. For example, values of an attribute expressing a length of a car can be transformed into intervals labelled as *short*, *medium* and *long*. Those new values are easer to comprehend and utilize by humans in their judgements. Of course, semantics of each of those values can be different for different people. It will also be dependent on particular cars represented in the data. For the purpose of a practical data analysis, however, it is often sufficient to apply a heuristic discretization technique to divide numerical attribute values into intervals representing meaningful qualitative symbols.

The discretization can be combined with dimensionality reduction, e.g. by using a discernibility-based discretization method described in [97] to compute a set of symbolic attributes that discern all objects in the data (or nearly all in the approximate case). In this approach a subset of attributes with a corresponding set of cuts is selected from a larger attribute set in a greedy fashion. It is done using a simple deterministic heuristic which starts with an empty set and iteratively adds the most promising attributes with corresponding cuts until the decision determination criterion is satisfied [71, 97]. Since the resulting set of discretized attributes discern all or sufficiently many[2] instances from different decision classes of the original decision system, it can be easily adjusted to become a desired type of a reduct (definitions of several types of reducts can be found in Section 2.3). For this purpose cuts that

[2]A desired number of discerned instances can be treated as a parameter that governs the approximation quality.

Algorithm 2: The calculation of a decision reduct from numerical data

Input: a decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$;
Output: a decision reduct $DR \subset A$ coupled with sets of cuts for each attribute from DR;

begin
  DR = empty list; `// an empty list of attributes`
  SC = empty list; `// an empty list of selected cuts`
  $CC = \emptyset$; `// a set of cut candidates`
  foreach $a \in A$ **do**
    Compute cut candidates CC_a for the attribute a using the guidelines from [97];
    $CC = CC \cup CC_a$;
    $SC[a] = \emptyset$;
  end
  $i = 1$; **while** *there are conflicts in* $\mathbb{S}'_d = (U, DR', d)$ **do**
    $Q_{max} = -\infty$;
    foreach *cut* $\in CC$ **do**
      $Q(cut)$ = quality of *cut*;
      if $Q(cut) > Q_{max}$ **then**
        $Q_{max} = Q(cut)$;
        $best_{cut} = cut$;
        $best_a$ = attribute a which corresponds to $best_{cut}$;
      end
    end
    $DR[i] = best_a$;
    $SC[a] = SC[a] \cup best_{cut}$;
  end
  foreach $a \in DR$ **do**
    if *there are no conflicts in* $\mathbb{S}'_d = (U, (DR \setminus \{a\})', d)$ **then**
      $DR = DR \setminus \{a\}$;
      $SC[a] = \emptyset$;
    end
  end
  return DR and SC;
end

are abundant need to be eliminated. Therefore such a method can be viewed as a simultaneous supervised discretization and computation of decision reducts [71, 73]. This approach to the dimensionality reduction does not only boost the construction of RBS, but is also helpful in identification of truly relevant local features. For those reasons it was used in all experiments with RBS on numerical data presented in this dissertation (see Sections 5.1 and 5.2). Algorithm 2 shows the procedure for classical reducts which in this case are understood as irreducible sets of *discretized* attributes

that discern all objects from different decision classes[3] [71]. The algorithm assumes that due to the presence of numerical attributes there is no inconsistency in the original data table (i.e. there are no indiscernible objects).

In Algorithm 2, DR' denotes a set of attributes from DR discretized using the corresponding cuts from the list SC. To facilitate computations for high dimensional data some randomness can be introduced to the generation of candidate cut. In this way the algorithm can be employed for finding a diverse set of good quality decision reducts [71, 69]. This approach was used in the extension to RBS which is discussed in Section 4.3.4. The resulting set of attributes can also be a super-reduct by skipping the attribute elimination phase in order to capture more, potentially important similarity aspects.

Because a class or a type of an object may have a significant impact on its similarity assessments to other objects in data [42, 159, 160], different sets of important features need to be extracted for different decision classes. For this reason, in a case when a decision attribute in data has more than two values, the discretization and attribute reduction in RBS need to be performed separately for each decision class, using the one-vs-all approach.

In the second step, higher-level features that are relevant for the judgement of similarity are derived from data using a rule mining algorithm. Each of those features is defined by the characteristic function of the left-hand side (the antecedent) of a rule. In RBS, two types of rules are generated – decision rules (see Definition 2.3) that form a set of candidates for relevant positive features, and inhibitory rules (see Definition 2.4) which are regarded as relevant distinctive features. Depending on a type of a rule, the corresponding feature can be useful either as an argument for or against the similarity to a matching object.

The induction of rules in RBS may be treated as a process of learning aggregations of local similarities from data. Features defined by antecedents of the rules express higher-level properties of objects. For instance, a characteristic indicating that a car is big may be expressed using a formula:

$$car\_length = high \wedge car\_width = high \wedge car\_height = high \ .$$

The feature defined in this way approximates the concept of a big car. Such a concept is more likely to be used by a person who assesses the similarity between two cars in a context of their appearance, than the exact numerical values of lengths, widths and heights. It can be noticed, for example, when people are explaining why they think that two objects are similar. It is more natural to say that two cars are similar because they are both big, than it is to say that one of them has $5,034mm$ length, $1,880mm$ width and $1,438mm$ height; the other is $5,164mm$ long, $1,829mm$ wide and $1,415mm$ high, and the differences in the corresponding parameters are small.

The choice of the higher-level features in RBS is not unique. Different heuristics for computation of reducts and different parameter settings of rule induction algorithms lead to the construction of different feature sets. As a consequence, the corresponding similarity approximation space changes along with the representation of the objects. The new representation may define a family of indiscernibility classes

[3] A discretized attribute corresponds to a pair consisting of the original attribute and a set of cuts that define nominal values (intervals).

which is better fitted to the approximation of similarities to particular objects. In this context, it seems trivial to say that some approximation spaces are more suitable for approximating the similarities than others. Therefore the problem of learning the similarity relation in RBS is closely related to searching for a relevant approximation space [133, 134] (see also the discussion in Section 2.2).

More formally, let $F^{+}_{(i)}$ and $F^{-}_{(i)}$ be the sets of binary features derived from the decision and the inhibitory rules (see Definitions 2.3 and 2.4), respectively, generated for i-th decision class:

$$\begin{aligned} F^{+}_{(i)} &= \left\{\phi : \Big(\phi \rightarrow (d = i)\Big) \in RuleSet_i\right\}, \\ F^{-}_{(i)} &= \left\{\phi : \Big(\phi \rightarrow \neg(d = i)\Big) \in RuleSet_i\right\}. \end{aligned}$$

$RuleSet_i$ is a set of rules derived from a reduct DR_i associated with the i-th decision class. The rule set may be generated using any rule mining algorithm but it is assumed, that if not stated otherwise, $RuleSet_i$ consists of rules that are true in $\mathbb{S}$ (their *confidence factor* is equal to 1 – see Section 2.1.3) and cover all available training data, i.e. for every $u \in U$ there exists $\pi \in RuleSet_i$ such that $u \vDash lhs(\pi)$. Moreover, for efficiency in practical applications of the model it may be necessary to require that the generated sets of rules $RuleSet_i$ be minimal. It means that there is no rule $\pi \in RuleSet_i$ that could be removed without reducing the set of covered objects or, in other words, for every $\pi \in RuleSet_i$ there exists $u \in lhs(\pi)(U)$ which is not covered by any other rule from $RuleSet_i$.

A feature ϕ is also a decision logic formula, i.e. a conjunction of descriptors defined over discretized attributes, that corresponds to an antecedent of some rule (see the notation introduced in Section 2.1.3). We will say that an object u, described in a decision system $\mathbb{S} = (U, A)$, has a feature ϕ iff $u \vDash \phi$. A set of all objects from U that have the feature ϕ (the meaning of ϕ in $\mathbb{S}$) will be denoted by $\phi(U)$.

In RBS a similarity relation is approximated by means of approximating *multiple concepts* of being similar to a specific object. In the rough set setting, a similarity to a specific object is a well-defined concept. In the proposed model, it consists of those object from U which share with u at least one feature from the set $F^{+}_{(i)}$, where i is *assumed* to be the decision class of u ($d(u) = i$):

$$SIM_{(i)}(u) = \bigcup_{\phi \in F^{+}_{(i)} \wedge u \vDash \phi} \phi(U)$$

Analogically, the approximation of the dissimilarity to u is a set of objects from U which have at least one feature from $F^{-}_{(i)}$ that is *not in common* with u:

$$DIS^{0}_{(i)}(u) = \bigcup_{\phi \in F^{-}_{(i)} \wedge u \nvDash \phi} \phi(U)$$

For convenience, the set of objects that have at least one feature from $F^{-}_{(i)}$ that is *in common* with u will be denoted by:

$$DIS^{1}_{(i)}(u) = \bigcup_{\phi \in F^{-}_{(i)} \wedge u \vDash \phi} \phi(U)$$

To abbreviate the notation only $SIM(u)$ and $DIS(u)$ will be written when the decision for an object u is known:

$$SIM(u) = SIM_{(d(u))}(u); \qquad DIS(u) = DIS^0_{(d(u))}(u)$$

It is worth noticing that within the theory of rough sets the set $SIM(u)$ can be seen as an outcome of an uncertainty function $SIM : U \to \mathbb{P}(U)$ (see Definition 2.5). A proof of this fact is quite trivial. From the definition of the set $SIM(u)$ it follows that $u \in SIM(u)$. Moreover, if $u_1 \in SIM(u_2)$, then there exists $\phi \in F^+_{(d(u_2))}$ such that $u_1 \in \phi(U) \wedge u_2 \vDash \phi$. If so, then $u_1 \vDash \phi$, thus $d(u_1) = d(u_2)$ and $u_2 \in SIM(u_1)$.

Analogically, the set $DIS(u)$ is an outcome of a function $DIS : U \to \mathbb{P}(U)$ which can be seen as an opposite of SIM. The function SIM induces a tolerance relation in U, whereas DIS induces a relation that can be called an *intolerance relation*. From the definition, $\forall_{u \in U} u \notin DIS(u)$, i.e. the relation induced by DIS is anti-reflexive. Moreover, this relation is asymmetric since for every $u_1, u_2 \in U$, if $u_1 \in DIS(u_2)$ then $u_2 \notin DIS^0_{(d(u_2))}(u_1)$.

The functions SIM and DIS are used for the approximation of the similarity and the dissimilarity to objects from U. In the RBS model, the assessment of a degree in which an object u_1 is similar and dissimilar to u_2 is done using two functions:

$$Similarity(u_1, u_2) \;=\; \mu_{sim}\big(u_1, SIM_{d(u_1)}(u_2)\big) \;=\; \frac{\big|SIM(u_1) \cap SIM_{d(u_1)}(u_2)\big|}{|SIM(u_1)| + C_{sim}}\,,$$

$$Dissimilarity(u_1, u_2) \;=\; \hat{\mu}_{dis}\big(u_1, DIS^1_{d(u_1)}(u_2)\big) \;=\; \frac{\Big|DIS(u_1) \cap DIS^1_{d(u_1)}(u_2)\Big|}{|DIS(u_1)| + C_{dis}}\,.$$

In the above formulas C_{sim} and C_{dis} are positive constants which can be treated as parameters of the model. The function $\mu_{sim} : U \times \mathbb{P}(U) \to [0, 1)$ can be seen as a membership function from the rough set theory (see Definition 2.6). It measures a degree in which an object u_1 fits to the concept of the similarity to u_2. The function $\hat{\mu}_{dis} : U \times \mathbb{P}(U) \to [0, 1)$ may be regarded as an *anti-membership* function since it measures a degree in which u_1 is not similar to u_2 (i.e. is dissimilar to u_2). It is also worth noticing that if the assumptions regarding the consistency and the coverage of the utilized rules are true, then for every $u \in U$, $|SIM(u)| > 0$ and $|DIS(u)| > 0$, and the functions $Similarity$ and $Dissimilarity$ are well-defined for every pair $(u_1, u_2) \in U \times \Omega$, even in a case when $C_{sim} = C_{dis} = 0$.

The similarity function of the RBS model combines values of $Similarity$ and $Dissimilarity$ for a given pair of objects. It can be expressed as:

$$Sim_{RBS}(u_1, u_2) = F\Big(Similarity(u_1, u_2), Dissimilarity(u_1, u_2)\Big) \tag{4.5}$$

where $F : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ can be any function that is monotonically increasing with regard to its first argument (i.e. a value of $Similarity$) and monotonically decreasing with regard to its second argument (a value of $Dissimilarity$). One example of such a function can be:

$$Sim_{RBS}(u_1, u_2) = \frac{Similarity(u_1, u_2) + C}{Dissimilarity(u_1, u_2) + C} \tag{4.6}$$

where $C > 0$ is a small constant, which is introduced to avoid division by zero and to ensure that $Sim_{RBS}(u_1, u_2) = 1$ for u_1, u_2 which are neither similar nor dissimilar (i.e. $Similarity(u_1, u_2) = Dissimilarity(u_1, u_2) = 0$). In this particular form the RBS similarity function was used in experiments described in Sections 5.1 and 5.2.

Alternatively, a similarity degree in RBS could also be expressed as a simple difference between the similarity and dissimilarity of two objects, as in the case of Tversky's model:

$$Sim'_{RBS}(u_1, u_2) = Similarity(u_1, u_2) - Dissimilarity(u_1, u_2) \tag{4.7}$$

In this form, the RBS function takes values between -1 and 1, with its neutral value equal 0. An advantage of this function is that it does not need the additional constant C. It can be easily shown that all the mathematical properties of Sim_{RBS}, which are discussed in Section 4.3.3, are independent of the exact form of the function F as long as the requirement regarding its monotonicity is met.

Depending on the type and parameters of a rule mining algorithm utilized for the creation of the feature sets $F^+_{(i)}$ and $F^-_{(i)}$, the sets $SIM(u)$ and $DIS(u)$ can have different rough set interpretations (Figure 4.5). If all the rules are true in $\mathbb{S}$, then $SIM(u)$ and $DIS(u)$ would be equivalent to lower approximations of the concepts of similarity and dissimilarity to u in U, respectively. Otherwise, if the rules with a lower confidence coefficient were allowed, $SIM(u)$ and $DIS(u)$ would correspond to upper approximations of the similarity and the dissimilarity to u. Their properties and granulation may be treated as parameters of the model. In applications they can be tuned to boost the quality of the induced relation. This tuning process can be regarded as searching for the optimal approximation space (see Section 2.2.2).

Figure 4.5 shows a simplified graphical interpretation of the RBS model. The grey area in the picture represents a concept of similarity to object u_1 from the decision class $d(u_1)$. The rectangles inside this region correspond to an approximation of the concept of being similar to u_1. They are defined by indiscernibility classes of training objects that share at least one feature from $F^+_{(d(u_1))}$ with u_1. Analogically, the rectangles outside the decision class approximate the concept of the dissimilarity to u_1 and they contain instances from the set $DIS^0_{d(u_1)}(u_1)$. The local similarity value of u_2 to u_1 in this example would be calculated as a ratio between a fraction of the similarity approximation shared by u_1 and u_2, and a fraction of the dissimilarity approximation which is characteristic only to u_2. In Figure 4.5, areas corresponding to those fractions are highlighted in blue and red, respectively.

The function Sim_{RBS} can be employed for the classification of objects from unknown decision classes as it only uses information about the class of the first object from the pair. New objects can be classified in a cased-based fashion, analogically to the *k-nearest neighbors* algorithm. Exemplary similarity-based classification functions are presented in Section 3.1.3.

4.3.3 Properties of the Rule-Based Similarity function

To illustrate the evaluation of the similarity in RBS, let us consider the decision system from Table 4.3. Assume that we want to evaluate the similarity of *Ford Mustang* to *New_Car* in a context of their appearance, which is judged by a given

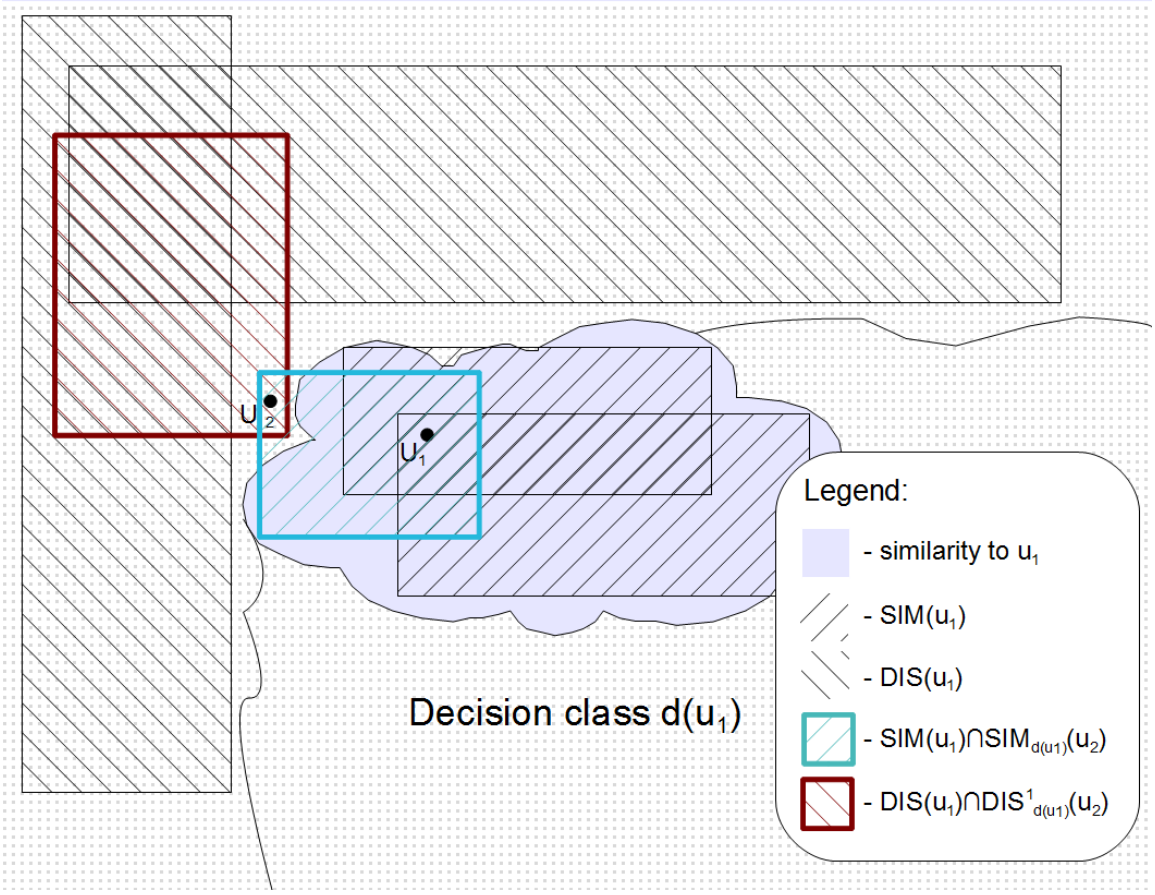


Figure 4.5: A graphical interpretation of an approximation of similarity to a single object in RBS.

person. We know preferences of this person regarding cars (the classes of objects) from our decision table but we have no information regarding the classification of *New_ Car*. During the construction of the RBS model, the data set describing the selected cars was discretized and some consistent decision rules[4] were induced for each of the two possible classes. Since the decision for *Ford Mustang* is *Nice*, we choose the positive features from the rules pointing at this class (i.e. rules in a form of $\phi \rightarrow Nice$). The negative features are chosen among the rules indicating the *notNice* decision.

Suppose that from the set of antecedents of the rules induced for the decision *Nice*, two were matching *New_ Car*: ϕ_1^+ and ϕ_4^+. Additionally, there was one feature derived from a rule classifying objects as *notNice*, that matched the tested car: ϕ_1^-. From the decision table we know that *Ford Mustang* has in common with *New_ Car* only the feature ϕ_1^+, so this feature is an argument for their similarity. In addition, the feature ϕ_1^- does not match *Ford Mustang* therefore this feature provide an argument for dissimilarity of the compared cars. Although the rule ϕ_4^+ does not match *Ford Mustang*, it is not considered as an argument for the dissimilarity of the two cars

[4]Since there are only two decisions, inhibitory rules for one class correspond to decision rules for the other.

Table 4.3: An exemplary decision table displaying one's preferences regarding general appearance of selected cars. $F^{+}_{Nice} = \{\phi^{+}_1, \phi^{+}_2, \phi^{+}_3, \phi^{+}_4\}$ and $F^{-}_{Nice} = \{\phi^{-}_1, \phi^{-}_2, \phi^{-}_3, \phi^{-}_4\}$.

Object:	ϕ^{+}_1	ϕ^{+}_2	ϕ^{+}_3	ϕ^{+}_4	ϕ^{-}_1	ϕ^{-}_2	ϕ^{-}_3	ϕ^{-}_4	Decision
Ford Mustang	1	0	1	0	0	0	0	0	*Nice*
Toyota Avensis	0	0	0	0	1	1	0	1	*notNice*
Audi A4	0	0	0	0	1	0	1	0	*notNice*
Porsche Carrera	0	1	0	1	0	0	0	0	*Nice*
Mercedes S-Class	0	0	0	0	0	1	0	1	*notNice*
Chevrolet Camaro	0	1	1	0	0	0	0	0	*Nice*
Volkswagen Passat	0	0	0	0	0	1	1	0	*notNice*
Mitsubishi Eclipse	1	0	1	1	0	0	0	0	*Nice*

because the features from the set F^{+}_{Nice} may only become arguments for the similarity. Since two out of three cars which match to the features of *Ford Mustang* have the feature ϕ^{+}_1 and three out of four cars with decision *notNice* have the feature ϕ^{-}_1, if we set $C_{sim} = C_{dis} = 1$, the RBS value equals:

$$\begin{aligned} Sim_{RBS}(FordMustang, New\_Car) &= \Big(\frac{2}{3 + C_{sim}} + C\Big) \Big/ \Big(\frac{3}{4 + C_{dis}} + C\Big) \\ &= \Big(\frac{2 + 4C}{4}\Big) \Big/ \Big(\frac{3 + 5C}{5}\Big) = \frac{5 + 10C}{6 + 10C} \ . \end{aligned}$$

For a very small value of C we get the value $\approx \frac{5}{6}$. Since this value is lower than 1, *Ford Mustang* should be considered dissimilar to *New_ Car*.

The RBS model shares many properties with Tversky's contrast model of the similarity. In both models the evaluation of the similarity is seen as a feature matching process. Objects from the data are represented by sets of qualitative features rather than by vectors in an attribute space [159]. Furthermore, both models consider features as possible arguments for or against the similarity and aggregate those arguments during the similarity assessment.

The construction of RBS makes the resulting model flexible and strongly data-dependent. As in the contrast model, in RBS the similarity function is likely to be not symmetric, especially when the compared objects are from different decision classes. Moreover, in a case of inconsistency of a data set (see Definition 2.2), a relation induced using the RBS similarity function may be even not reflexive. This fact is in accordance with the main feature of the similarity for the classification (Definition 3.1). It also reflects a phenomena, that availability of information about decision classes (types or predefined labels) of examined stimuli impacts human judgements of the similarity [41, 42].

The similarity functions of RBS and the contrast model also have in common a number of mathematical properties, such as the maximality of marginal values and the monotonicity with regard to the inclusion of the feature sets:

Proposition 4.3.1. *Let* $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ *be a consistent decision system,* $U \subseteq \Omega$ *and let* $Sim_{RBS} : U \times \Omega \to \mathbb{R}$ *be a similarity function of the RBS model, constructed*

for $\mathbb{S}_d$ using rules that are true in $\mathbb{S}_d$ and cover all objects from U. The following inequity holds for every $u \in U$ and $u' \in \Omega$:

$$Sim_{RBS}(u,u) \geq Sim_{RBS}(u,u') \ .$$

Proof. To prove this inequity it is sufficient to show that $Similarity(u,u)$ is maximal and $Dissimilarity(u,u) = 0$ for every $u \in U$. Since the RBS model is constructed from rules that cover all objects from U, $|SIM(u)| > 0$ and for any $X \subseteq U$ we have:

$$Similarity(u,u) = \frac{|SIM(u) \cap SIM(u)|}{|SIM(u)| + C_{sim}} \geq \frac{|SIM(u) \cap X|}{|SIM(u)| + C_{sim}} \ .$$

Analogically, $|DIS(u)| > 0$ and since the utilized rules are true in $\mathbb{S}_d$, $DIS^1_{d(u)}(u) = \emptyset$. If so, then

$$Dissimilarity(u,u) = \frac{\left|DIS(u) \cap DIS^1_{d(u)}(u)\right|}{|DIS(u)| + C_{dis}} = 0 \ .$$

□

Proposition 4.3.2. *Let $\mathbb{S}_d = \left(U, A \cup \{d\}\right)$ be a consistent decision system, $U \subseteq \Omega$ and let $Sim_{RBS} : U \times \Omega \to \mathbb{R}$ be a similarity function of the RBS model, constructed for $\mathbb{S}_d$ using rules that cover all objects from U. In addition, let us consider objects $u \in U$ and $u', u'' \in \Omega$, such that u' and u'' are represented by feature sets $\{\Phi^+_{(d(u))}, \Phi^-_{(d(u))}\}$ and $\{\Psi^+_{(d(u))}, \Psi^-_{(d(u))}\}$, respectively. The following implication holds for every $u \in U$:*

$$\left(\Phi^+_{(d(u))} \supseteq \Psi^+_{(d(u))} \wedge \Phi^-_{(d(u))} \subseteq \Psi^-_{(d(u))}\right) \Rightarrow Sim_{RBS}(u,u') \geq Sim_{RBS}(u,u'') \ .$$

Proof. To prove the above implication it is sufficient to show that for objects considered in the proposition we have $Similarity(u,u') \geq Similarity(u,u'')$ and $Dissimilarity(u,u') \leq Dissimilarity(u,u'')$.

Let us consider the sets $\Phi^+_{(d(u))}$ and $\Psi^+_{(d(u))}$:

$$\begin{aligned} \Phi^+_{(d(u))} \supseteq \Psi^+_{(d(u))} \ &\Rightarrow \ SIM_{(d(u))}(u') \supseteq SIM_{(d(u))}(u'') \\ &\Rightarrow \ \left|SIM(u) \cap SIM_{(d(u))}(u')\right| \geq \left|SIM(u) \cap SIM_{(d(u))}(u'')\right| \end{aligned}$$

This and the fact that $\forall_{u \in U} |SIM(u)| > 0$ implies that $Similarity(u,u') \geq Similarity(u,u'')$.

Analogically, if $\Phi^-_{(d(u))} \subseteq \Psi^-_{(d(u))}$ then $DIS^1_{d(u)}(u') \subseteq DIS^1_{d(u)}(u'')$ and as a consequence $Dissimilarity(u,u') \leq Dissimilarity(u,u'')$. □

Proposition 4.3.3. *Let $\mathbb{S}_d = \left(U, A \cup \{d\}\right)$ be a consistent decision system, $U \subseteq \Omega$ and let $Sim_{RBS} : U \times \Omega \to \mathbb{R}$ be a similarity function of the RBS model, constructed for $\mathbb{S}_d$ using rules that are true in $\mathbb{S}_d$ and cover all objects from U. In addition, let us consider objects u, u', such that $d(u) = d(u') = i$ and u, u' are represented by feature sets $\{\Phi^+_{(i)}, \Phi^-_{(i)}\}$ and $\{\Psi^+_{(i)}, \Psi^-_{(i)}\}$, respectively. The following implication holds for any such $u, u' \in U$:*

$$\left(\Phi^+_{(i)} \supseteq \Psi^+_{(i)} \wedge \Phi^-_{(i)} \subseteq \Psi^-_{(i)}\right) \Rightarrow Sim_{RBS}(u,u') \leq Sim_{RBS}(u',u) \ .$$

Proof. It is sufficient to show that for all objects $u, u' \in U$ considered in the proposition, $Similarity(u, u') \leq Similarity(u', u)$ and $Dissimilarity(u, u') \geq Dissimilarity(u', u)$.

The second inequity is trivial due to the fact that $d(u) = d(u')$ and the rules are true in $\mathbb{S}_d$. In such a case $DIS^1(u) = DIS^1(u') = \emptyset$ and $Dissimilarity(u, u') = Dissimilarity(u', u) = 0$. To show the validity of the first inequity let us consider $u, u' \in U$ described by feature sets $\Phi^+_{(i)}$ and $\Psi^+_{(i)}$, respectively. We have:

$$\begin{aligned} \Phi^+_{(i)} \supseteq \Psi^+_{(i)} &\Rightarrow SIM(u) \supseteq SIM(u') \\ &\Rightarrow SIM(u) \cap SIM(u') = SIM(u') \quad \text{and} \\ &\quad |SIM(u)| \geq |SIM(u')| \ . \end{aligned}$$

If so, then:

$$\begin{aligned} Similarity(u, u') &= \frac{|SIM(u) \cap SIM(u')|}{|SIM(u)| + C_{sim}} = \frac{|SIM(u')|}{|SIM(u)| + C_{sim}} \\ &\leq \frac{|SIM(u')|}{|SIM(u')| + C_{sim}} = Similarity(u', u) \ . \end{aligned}$$

That concludes the proof. □

The next proposition shows that the RBS similarity function is suitable for constructing approximations of similarity relations in the context of classification. Fundamental properties of such relations were discussed in Section 3.1.2. However, before we can formulate this proposition we first need to prove a simple lemma:

Lemma 4.3.4. *Let Π be a set of decision rules generated for a consistent decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ and let Π_1, Π_2 denote two subsets of Π. Additionally, let $supp(\Pi_1) = \bigcup_{\pi \in \Pi_1} lhs(\pi)(U)$ and $supp(\Pi_2) = \bigcup_{\pi \in \Pi_2} lhs(\pi)(U)$. If Π covers all objects from U and is minimal in U, then*

$$supp(\Pi_1) \subseteq supp(\Pi_2) \Leftrightarrow \Pi_1 \subseteq \Pi_2 \ .$$

Proof. The implication $\Pi_1 \subseteq \Pi_2 \Rightarrow supp(\Pi_1) \subseteq supp(\Pi_2)$ is trivial. To prove the second implication, for a moment let us assume that the conditions from Lemma 4.3.4 are met and $supp(\Pi_1) \subseteq supp(\Pi_2)$ but there exists a rule $\pi \in \Pi_1$ such that $\pi \notin \Pi_2$. In such a case, $supp(\{\pi_1\}) \subseteq supp(\Pi_1) \subseteq supp(\Pi_2)$, so $\forall_{u \in lhs(\pi)(U)} \exists_{\pi' \in \Pi_2} u \vDash lhs(\pi')$. This, however, contradicts with the assumption that Π is minimal. □

A direct consequence of Lemma 4.3.4 is that $supp(\Pi_1) = supp(\Pi_2) \Leftrightarrow \Pi_1 = \Pi_2$.

In the following proposition there will be an additional assumption regarding the sets of rules $RuleSet_i$ used in the construction of the RBS model. Namely, apart from the consistency, coverage and minimality of the rule sets, it will be assumed that each $RuleSet_i$ is sufficiently rich to ensure the uniqueness of a representation by the sets of new features of all objects which are discernible in the original decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$. More formally, we will assume that for every $u, u' \in U$ represented by new feature sets $\{\Phi^+_{(i)}, \Phi^-_{(i)}\}$ and $\{\Psi^+_{(i)}, \Psi^-_{(i)}\}$, respectively, $u' \notin [u]_A \Leftrightarrow (\Phi^+_{(i)} \neq \Psi^+_{(i)} \vee \Phi^-_{(i)} \neq \Psi^-_{(i)})$. This property corresponds to the solvability assumption in Tversky's contrast model [159]. It is worth noticing that for any consistent decision

system $\mathbb{S}_d$ (see Definition 2.2) it is always possible to construct sets $RuleSet_i$ that meet all of the above requirements. In the simplest case, it is sufficient to take the rules whose predecessors correspond to descriptions of indiscernibility classes in $\mathbb{S}_d$ and successors point out the corresponding decisions.

Proposition 4.3.5. *Let τ be a similarity relation in a context of classification in a universe Ω. Additionally, let $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$ be a consistent decision system, $U \subseteq \Omega$ and let $Sim_{RBS} : U \times \Omega \to \mathbb{R}$ be a similarity function of the RBS model, constructed for $\mathbb{S}_d$ using rule sets, which have the properties of consistency, coverage, minimality and uniqueness of a representation. The function Sim_{RBS} is a proper similarity function for the relation τ within the set U.*

Proof. Let us denote by $\tau_{(\epsilon)}^{Sim_{RBS}}$ a set of all pairs $(u, u') \in U \times U$ for which $Sim_{RBS}(u, u') \geq \epsilon$. To show that the function Sim_{RBS} has the property of being a proper similarity function (Definition 3.2) for the relation τ within the set U we will give values of ϵ_1 and ϵ_2 such that for any $u, u' \in U$ we have:

$$Sim_{RBS}(u, u') \geq \epsilon_1 \;\Rightarrow\; (u, u') \in \tau \tag{4.8}$$

$$(u, u') \in \tau \;\Rightarrow\; Sim_{RBS}(u, u') \geq \epsilon_2 \tag{4.9}$$

and the sets $\tau_{(\epsilon_1)}^{Sim_{RBS}}$ and $U \setminus \tau_{(\epsilon_2)}^{Sim_{RBS}}$ are not empty.

We will start the proof by showing that if $Sim_{RBS}(u, u') = F\big(Similarity(u, u'), Dissimilarity(u, u')\big)$ for F that is increasing with regard to its first argument and decreasing with regard to the second, then the implication 4.8 is true for $\epsilon_1 = F(sim_{max}, 0)$, where $sim_{max} = \max\limits_{u \in U}\big(Similarity(u, u)\big)$. In particular, we will show that $Sim_{RBS}(u, u') \geq F(sim_{max}, 0) \Leftrightarrow (u' \in [u]_A \wedge u \in U_{max})$, where $U_{max} = \{u \in U : u = \operatorname{argmax}\limits_{u \in U} |SIM(u)|\}$.

Since all utilized rules are consistent and they cover all objects from U, for any $u, u' \in U$ we have $Dissimilarity(u, u') = 0 \Leftrightarrow d(u) = d(u')$. Moreover, due to the fact that $\mathbb{S}_d$ is consistent and the utilized rules uniquely represent the objects from U, for any $u \in U$ and $u' \in [u]_A$ we have $SIM(u') = SIM(u)$. If so, then $u \in U_{max} \Rightarrow [u]_A \subseteq U_{max}$ and

$$Similarity(u, u') = Similarity(u', u) = \frac{|SIM(u)|}{|SIM(u)| + C_{sim}} \ .$$

Thus, the inequity $Sim_{RBS}(u, u') \geq F(sim_{max}, 0)$ holds for every pair (u, u') such that $u \in U_{max}$ and $u' \in [u]_A$.

On the other hand, let us imagine that there exist objects $u, u' \in U$ such that $u \notin U_{max} \vee u' \notin [u]_A$ and $Sim_{RBS}(u, u') \geq F(sim_{max}, 0) \vee Sim_{RBS}(u', u) \geq F(sim_{max}, 0)$ (or, equivalently, $Similarity(u, u') \geq sim_{max} \vee Similarity(u', u) \geq sim_{max}$). If $u \notin U_{max}$ but $u' \in [u]_A$ we get an inconsistency, because all $u \in U$ for which $u' \in [u]_A$ and $Similarity(u, u')$ is maximal, by definition must belong to U_{max}. Now, if it is true that $u \in U_{max} \wedge u' \notin [u]_A$ and $Similarity(u, u') \geq sim_{max}$, then we have:

$$\begin{aligned} Similarity(u, u') \geq Similarity(u, u) \;&\Leftrightarrow\; \big|SIM(u) \cap SIM_{d(u)}(u')\big| \geq |SIM(u)| \\ &\Leftrightarrow\; d(u) = d(u') \wedge SIM(u) = SIM(u') \ . \end{aligned}$$

That also results in an inconsistency because, based on the assumption regarding the minimality of the rule sets and Lemma 4.3.4, the objects u and u' must have the same representation by new features, and thus $(u, u') \in IND_A$ (by the uniqueness of a representation). Hence, the only possibility left is that $u \in U_{max} \wedge u' \notin [u]_A$ and $Similarity(u', u) \geq sim_{max}$. In such a case we would have:

$$\begin{aligned} \frac{\left|SIM(u') \cap SIM_{d(u')}(u)\right|}{|SIM(u')| + C_{sim}} \geq sim_{max} \quad &\Leftrightarrow \quad d(u) = d(u') \wedge SIM(u') \subseteq SIM(u) \wedge \\ &\quad |SIM(u')| \geq |SIM(u)| \\ &\Leftrightarrow \quad SIM(u) = SIM(u') \ , \end{aligned}$$

which again contradicts with the assumption about the uniqueness of a representation and proves that $Sim_{RBS}(u, u') \geq F(sim_{max}, 0) \Leftrightarrow (u' \in [u]_A \wedge u \in U_{max})$. Since a similarity relation in the context of a classification is assumed to be reflexive, it shows that the implication 4.8 is true for $\epsilon_1 = F(sim_{max}, 0)$. Moreover, due to the fact that U is finite, the maximum value of the function $Similarity$ has to be taken by at least one pair $(u, u') \in U \times U$, and thus $\tau_{(\epsilon_1)}^{Sim_{RBS}} \neq \emptyset$.

To show that there exists ϵ_2 for which the implication 4.9 is true we will use the fact that τ is assumed to have the main feature of the similarity for the classification (see Definition 3.1). As we already noticed, due to the consistency and coverage of the utilized rules we have $Dissimilarity(u, u') = 0 \Leftrightarrow d(u) = d(u')$, and $d(u) \neq d(u') \Rightarrow Similarity(u, u') = 0$. If so, then for $\epsilon_2 = F(0, 0)$ we get $\tau_{(\epsilon_2)}^{Sim_{RBS}} \supseteq_U IND_{\{d\}} \supseteq_U \tau$. Moreover, since $Dissimilarity(u, u') > 0$ for any pair $(u, u') \in U \times U$ such that $d(u) \neq d(u')$, we have $U \setminus \tau_{(\epsilon_2)}^{Sim_{RBS}} \neq \emptyset$. Thus it is sufficient to take $\epsilon_2 = F(0, 0)$. □

4.3.4 Rule-Based Similarity for high dimensional data

In the Rule-Based Similarity model the notion of decision reduct is used for finding a concise set of attributes which can serve as building blocks for constructing higher-level features. Nevertheless, it has been noted that a single reduct may fail to capture all critical aspects of the similarity in a case when there are many important "raw" attributes. To overcome this problem, an extension to RBS called Dynamic Rule-Based Similarity (DRBS) was proposed [65, 67]. The main aim of the DRBS model is to extend the original model by taking into consideration a wider spectrum of possibly important aspects of the similarity.

During construction of the DRBS model, many independent sets of rules are generated from heterogeneous subsets of attributes. In this way, the resulting higher-level features are more likely to cover the factors that can influence similarity or dissimilarity of objects (the positive and negative feature sets) from a domain under scope. Within the model, the attributes that are used to induce the rules are selected by computation of multiple decision reducts from random subsets of data. This method can be seen as an analogy to the Random Forest algorithm [20], in which multiple decision trees are constructed. In DRBS however, the rules derived in this manner are not directly employed for classification but they are utilized to define multiple RBS similarity functions. Those local models are then combined in order to construct a single function which can yield a better approximation of a similarity relation in the context of a classification.

Algorithm 3: The computation of (ϵ, δ)-dynamic reducts in DRBS

Input: a decision system $\mathbb{S}_d = \big(U, A \cup \{d\}\big)$;
a parameter $NoOfAttr << |A|$;
parameters $\epsilon, \delta \in [0, 1)$;
integers $MaxDDR, MaxTry, NSets$;
Output: a set of (ϵ, δ)-dynamic reducts DDR_{set};

```
1  begin
2      DDR_set = ∅;
3      i = 0;
4      while |DDR_set| < MaxDDR ∧ i < MaxTry do
5          Randomly draw NoOfAttr attributes from A and construct A' ⊂ A,
           |A'| = NoOfAttr;
6          Compute a decision reduct DR of S'_d = (U, A', d);
7          k = 0;
8          for j = 1 to NSets do
9              Randomly draw ⌊(1 − ε) · |U|⌋ objects from U (without repetition)
               and create S''_d = (U', DR, d);
10             if DR ∈ RED(S''_d) then
11                 k = k + 1;
12             end
13         end
14         if k/NSets > 1 − δ then
15             DDR_set = DDR_set ∪ {DR};
16         end
17         i = i + 1;
18     end
19     return DDR_set;
20 end
```

Although in all experiments described in this dissertation DRBS was implemented using the (ϵ, δ)-dynamic decision reducts [9, 11] (see Definition 2.12), any kind of an efficient dimensionality reduction technique, such as approximate reducts [136, 138] or decision bireducts [141] could be used (see the definitions in Section 2.3). The dynamic decision reducts, however, tend to be reliable even in a case when only a few hundreds of objects are available for the learning and thus are suitable for coping with the *few-objects-many-attributes* problem [64, 139].

Algorithm 3 shows an efficient procedure for computing (ϵ, δ)-dynamic decision reducts. Although the algorithm does not give any guarantee as to the number of returned dynamic reducts, in practical experiments with real-life data sets (see Section 5.2) it has always successfully generated a sufficient number of reducts for constructing a reliable DRBS model. Its advantage for the similarity learning is that it naturally adjusts the number of generated local RBS models to the available data. In particular, for reasonable values of ϵ and δ, the number of produced reducts for data sets describing objects with many important similarity aspects is likely to be higher than for those which describe simpler problems, characterised with fewer

potentially important features.

The DRBS similarity function combines values of the local similarity functions. Due to a partially randomized reduct construction process, the individual RBS models represent more independent aspects of the similarity. That in turn results in a better performance of their ensemble [126, 141, 169]. This particular characteristic makes the DRBS model akin to the Random Forest algorithm where the final classification is done by combining decisions of multiple decision trees, constructed from random subsets of attributes and objects [20]. Unlike in the Random Forest, however, the classification results which are based on DRBS do not lose their interpretability. For each tested object we can explain our decision by indicating the examples from our data set which were used in the decision-making process (i.e. the k most similar cases). Equation 4.10 shows a basic form of a DRBS similarity function which averages outputs of the N local RBS models:

$$Sim_{DRBS}(u_1, u_2) = \frac{1}{N} \cdot \sum_{j=1}^{N} \Big(Sim_{RBS}^{(j)}(u_1, u_2) \Big), \tag{4.10}$$

where $Sim_{RBS}^{(j)}(u_1, u_2)$ is the value of the RBS similarity function for the j-th decision reduct. This function can be easily modified to reflect relative importances of individual RBS models:

$$Sim_{wDRBS}(u_1, u_2) = \frac{\omega^{(j)} \cdot \sum_{j=1}^{N} \Big(Sim_{RBS}^{(j)}(u_1, u_2) \Big)}{\sum_{j=1}^{N} \omega^{(j)}}. \tag{4.11}$$

In the above equation, weights $\omega^{(j)}$ correspond to quality of RBS models, which can be estimated using some of the methods described in Section 3.1.4. For this purpose, usually a part of objects from a learning set needs to be held back as a validation set.

DRBS introduces a few new parameters to the similarity model, of which the most important are $NoOfAttr$ and $MaxDDR$. They both govern the process of randomized computation of reducts. The first one tells how many attributes are randomly drawn from data for computation of a single reduct. The second one sets maximal number of the reducts to be generated. Together, those parameters influence the thickness of a coverage of truly important similarity aspects. Knowing their values, it is possible to estimate a chance of an attribute to be considered for inclusion into at least one reduct and the expected number of its occurrences within the final set of reducts.

If by p_{attr} we denote the ratio between $NoOfAttr$ and the total number of attributes in data ($p_{attr} = \frac{NoOfAttr}{|A|}$), the occurrence probability of an attribute $attr$ in at least one reduct and the expected number of its occurrences are equal:

$$p(attr) = 1 - MaxDDR \cdot (1 - p_{attr})^{MaxDDR} \quad and \quad E(attr) = MaxDDR \cdot p_{attr},$$

respectively. In practice, these two quantities can be used to set reasonable values of $NoOfAttr$ and $MaxDDR$ for a given data set.

Another two important parameters are ϵ and δ which have a significant impact on properties of generated dynamic reducts. The higher ϵ and lower δ, the more

robust are the resulting dynamic decision reducts. However, too restrictive values of those parameters may cause a serious deterioration in a computational efficiency of the algorithm or even prevent its completion.

Alternatively, if instead of dynamic reducts, the new feature sets were defined using decision bireducts, many parameters of the DRBS model could be replaced by a single ratio that governs the generation of random permutations (for more details refer to [141]). In practical experiments with DRBS, however, only the approximations derived from dynamic decision reducts have been used so far.

4.3.5 Unsupervised Rule-based Similarity for textual data

The idea behind the RBS model can also be applied to carry out unsupervised similarity learning [70]. In particular, the RBS model was extended to facilitate an approximation of a semantic similarity of scientific articles.

The construction of the model starts with assigning concepts from a chosen knowledge base to a training corpus of documents. This can be done in an automatic fashion with the use of methods such as ESA [38] (see Section 4.2.4). The associations to the key concepts assigned to the documents can be transformed to binary features and therefore, are suitable to use with the contrast model of similarity. However, a direct application of this model would not take into consideration data-based relations between concepts from the knowledge base and a potentially different meaning of those relations for different documents. The problem of finding appropriate values of parameters of the Tversky's model would also remain unsolved. The proposed extension of RBS aims to overcome those issues [70]. It is called the *unsupervised RBS* model, since it can be seen as a continuation of the research on the similarity learning model for high dimensional data [67].

Let F be a set of all possible semantic features of texts from a corpus D and let F_i be a set of the most important concepts related to the document T_i, $F = \bigcup_{i=1}^{|D|} F_i$. The documents from D can be represented in an information system $\mathbb{S} = (D, F)$, as explained in Section 4.2.4. An example of such a system is shown in Table 4.4. We can say that two documents described by this table have a common feature if they both have value 1 in the corresponding column (e.g. the documents T_1 and T_2 have three common features: f_2, f_3 and f_{10}). The binary attributes in this system may correspond to tags assigned by experts or by discretizing numeric weights of ontological entities generated using methods such as ESA.

In many practical applications, the numerical values of an association strength between a concept and a document may be discretized into more than two intervals in order to precisely model their bond. In this case, it is reasonable to define a few binary features that represent consecutive intervals and remain dependent, in a sense that if a feature is "*highly related*" to a document then it is also "*weakly related*", but not the opposite. Such an approach is popular in Formal Concept Analysis [40] and allows to model a psychological phenomena that usually simpler objects are more similar to the more complex ones than the other way around.

In order to find out which combinations of independent concepts comprise the informative aspects of similarity, we could compute information reducts of $\mathbb{S}$ [110, 131] or, to obtain more compact and robust subsets of F, some form of approximate

information reducts [71, 138]. However, during the research on the unsupervised RBS model the information bireducts were proposed [70] in order to limit its bias toward common concepts and objects of negligible importance.

Information bireducts can be defined similarly to the decision bireducts (see Section 2.3.3), however their interpretation is slightly different.

Definition 4.1 (Information bireduct)**.**
Let $\mathbb{S} = (D, F)$ be an information system. A pair (B, X), where $B \subseteq F$ and $X \subseteq D$, is called an information bireduct, iff B discerns all pairs of objects in X and the following properties hold:

1. *There is no proper subset $C \subsetneq B$ such that C discerns all pairs of objects in X.*
2. *There is no proper superset $Y \supsetneq X$ such that B discerns all pairs of objects in Y.*

Just as in the case of decision bireducts, information bireducts do not allow any inconsistencies in X. In a context of information bireducs, however, consistence is understood as an ability to distinguish between any pair of objects in the selected set.

It is interesting to compare information bireducts with templates studied in the association rule mining [96, 101] or concepts known from the formal concept analysis [39, 40]. Templates aim at describing a maximum number of objects with the same (or similar enough) values on a maximum number of attributes. Similarly, concepts are defined as non-extendable subsets of objects that are indiscernible with respect to non-extendable subsets of attributes. On the other hand, information bireducts describe non-extendable subsets of objects that are discernible using irreducible subsets of attributes. The templates and concepts might be seen as corresponding to the most regular areas of data, while the information bireducts correspond to the most irregular, chaotic or one might even claim – the most informative data. Hence, information bireducts can be also called anti-templates or anti-concepts.

In a context of similarity learning, information bireducts can also be intuitively interpreted as artificial agents that try to assess the similarity between given objects. Each of such agents can be characterised by its experience and preferences. In a bireduct, the experience of an agent is explicitly expressed by the set X – the set of cases that the agent knows. The preferences of an agent are modelled in an information bireduct by the set of attributes which are the factors taken into account when the agent makes a judgement. Such an interpretation makes information bireducts become an interesting tool for constructing similarity models from data.

For each bireduct $BR = (B, X)$, $B \subseteq F$, $X \subseteq D$, we can define a commonality relation in D with regard to BR. One example of such a relation can be $\varsigma|_{BR}$ which is defined as follows:

$$(T_i, T_j) \in \varsigma|_{BR} \iff T_j \in X \wedge \left|F_{i|_{BR}} \cap F_{j|_{BR}}\right| \geq p, \tag{4.12}$$

where $p > 0$, $T_i, T_j \in D$ and $F_{i|_{BR}}$ is a representation of T_i restricted to features from B. Intuitively, two documents are in the commonality relation $\varsigma|_{BR}$ if and only if one of them is covered by the bireduct BR and they have at least p common concepts. The commonality class of a document T with regard to BR will be denoted by $I_{BR}(T)$ since it can be regarded as a specific type of an uncertainty function in the theory of

Table 4.4: An information system $\mathbb{S}$ representing a corpus of nine documents, with three exemplary bireducts.

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
T_1	1	1	1	0	0	0	0	0	1	1
T_2	0	1	1	1	1	0	1	0	0	1
T_3	1	1	0	0	0	0	1	0	0	0
T_4	0	0	0	0	1	0	0	1	0	0
T_5	1	0	1	0	1	1	0	0	0	0
T_6	1	0	1	0	0	0	0	0	0	0
T_7	0	1	1	0	0	1	1	1	0	0
T_8	0	0	0	0	1	1	1	1	1	0
T_9	1	1	0	0	1	0	0	0	1	0

Exemplary information bireducts:

$BR_1 = (\{f_2, f_3, f_8, f_9\}, \{T_1, T_2, T_3, T_4, T_6, T_7, T_8, T_9\})$

$BR_2 = (\{f_3, f_5, f_7, f_9\}, \{T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8, T_9\})$

$BR_3 = (\{f_1, f_2, f_5\}, \{T_2, T_3, T_5, T_6, T_7, T_8, T_9\})$

rough sets (see Definition 2.5). For instance, if we consider the information system from Table 4.4 and the commonality relation defined by the formula (4.12) with $p = 2$, then $I_{BR_1}(T_1) = \{T_1, T_2, T_7, T_9\}$ and $I_{BR_1}(T_5) = \emptyset$.

It is important to realize that a commonality of two objects is something conceptually different than indiscernibility. For example, the documents T_5 and T_6 are indiscernible with regard to the features from the bireduct BR_1 but they are not in the commonality relation since they have only one feature f_3 in common.

A similarity model needs to have a functionality which allows it to be applied for analysis of new documents. Typically, we would like to assess their similarity to the known documents (those available during the learning phase) in order to index, classify or assign them to some clusters. For this reason, in the definition of the commonality relation only T_j needs to belong to X. This also makes it more convenient to utilize information bireducts that explicitly define the set of reference cases for which the comparison with the new ones is well defined.

The commonality relation (4.12) can be used to locally estimate the real significance of arguments for and against the similarity of documents which are being compared. Those arguments, i.e. sets of higher-level features of documents, can be aggregated analogously to the case of the regular RBS similarity function (4.5). In particular, the similarity of T_i to T_j with regard to a bireduct BR can be computed using the following formula:

$$Sim_{BR}(T_i, T_j) = \frac{|I_{BR}(T_i) \cap I_{BR}(T_j)|}{|I_{BR}(T_i)| + C} - \frac{\left|\left(X \setminus I_{BR}(T_i)\right) \cap I_{BR}(T_j)\right|}{|X \setminus I_{BR}(T_i)| + C}. \tag{4.13}$$

As in the case of the functions *Similarity* and *Dissimilarity* of the RBS model (see Section 4.3.2), the constant $C > 0$ is added to avoid division by zero.

Since each information bireduct is a non-extendable subset of documents, coupled with an irreducible subset of features that discern them, it carries maximum information on a diverse set of reference documents. Due to this property, the utilization of bireducts nullifies the undesired effect which common objects (or usual features) would impose on sizes of the commonality classes and thus, on the similarity function value. Moreover, such a use of the information bireducts in combination

with the commonality relation (4.12) substitutes the need for manual tuning of additional parameters. Instead, the relative intersection size of the commonality classes locally expresses the relevance of arguments for similarity without a need for considering additional parameters. By analogy, the importance of arguments against the similarity is reflected by the relative size of a set that comprises those documents which are not in the commonality class of the first document and are sufficiently compliant with the second text.

Following the example from Table 4.4, the formula (4.13) can be used to compute the similarity between any two documents from $\mathbb{S}$ with regard to a chosen bireduct BR_i. For instance, for a very small c, $Sim_{BR_1}(T_1, T_2) \approx 3/4 - 0 = 0.75$, $Sim_{BR_1}(T_1, T_5) = 0 - 0 = 0$ and $Sim_{BR_1}(T_1, T_8) \approx 0 - 1/4 = -0.25$. It is worth noting that the proposed approach keeps the flexibility of the original RBS model and does not impose any properties on the resulting similarity function. Depending on the data and on the selection of $\tau|_{BR}$, the function Sim_{BR} may be not symmetric ($Sim_{BR_1}(T_2, T_1) \approx 0.8 \neq Sim_{BR_1}(T_1, T_2)$), and even not reflexive ($Sim_{BR_1}(T_3, T_3) = 0$). In this case the lack of the reflexivity is a consequence of the fact that $T_3 \notin X$, thus a meaningful assessment of the similarity to this document is not possible. This flexibility of the model makes it consistent with observations made by psychologists [159].

The utilization of information bireducts allows to conveniently model different aspects of similarity. By analogy to the initial experiments with decision bireducts [141], a set of information bireducts will cover much broader aspects of data than an equally sized set of the regular information reducts. This allows to capture approximate dependencies between features which could not be discovered using classical methods and may contribute to the overall performance of the model. The algorithm proposed in [141] for computation decision bireducts can be easily adjusted to the case of information bireducts (Algorithm 4). The randomization of the algorithm guarantees that its multiple executions will produce a diverse set of bireducts.

To robustly evaluate similarity of two documents the agents need to interact by combining their assessments. The simplest method of such an interaction is to average votes of all agents. In such a case, the final similarity of T_i to T_j can be computed using the following formula:

$$Sim(T_i, T_j) = \frac{\sum_k Sim_{BR_k}(T_i, T_j)}{\#extracted\ bireducts}. \tag{4.14}$$

For example, if for the information system $\mathbb{S}$ from Table 4.4 we consider information bireducts BR_1, BR_2 and BR_3, the final similarity of T_1 to T_2 would be equal to $Sim(T_1, T_2) = (0.75 + 0.05 + 0.3)/3 \cong 0.37$.

The design of such a similarity function is computationally feasible and does not require tuning of unintuitive parameters. It also guarantees that the resulting similarity function keeps the flexibility and psychologically plausible properties. Moreover, this kind of an ensemble significantly reduces the variance of similarity judgements in a case when the available data set changes over time (e.g. new documents are added to the repository) and increases model robustness.

However, some more sophisticated methods can also be employed for carrying out the interaction between the agents (bireducts), in order to improve performance in

Algorithm 4: The calculation of an information bireduct of $\mathbb{S} = (D, F)$

Input: an information system $\mathbb{S} = (D, F)$;
a random permutation $\sigma : \{1, ..., |D| + |F|\} \to \{1, ..., |D| + |F|\}$;
Output: an information bireduct (B, X), $B \subseteq F$, $X \subseteq D$;

```
begin
    B = F;
    X = ∅;
    for i = 1 to |D| + |F| do
        if σ(i) ≤ |F| then
            if B \ {F_σ(i)} discerns all pairs in X then
                B ← B \ {F_σ(i)}
            end
        end
        else
            if B discerns all pairs in X ∪ {T_σ(i)−K} then
                X ← X ∪ {T_σ(i)−K}
            end
        end
    end
    return (B, X);
end
```

a given task or to reduce similarity computation costs. For instance, properties of extracted bireducts can be used to select only those which will most likely contribute to the performance of the model. The considered properties may include, e.g. a number of selected features, a size of the reference document subset or an average intersection with other bireducts [141]. Using such statistics in combination with general knowledge about the data it is possible to decrease the number of bireducts required for making consistent similarity assessments.

4.3.6 Summary of the Rule-Based Similarity models

The construction of the RBS model makes it flexible and allows to apply it in many object domains. By its design, the model tries to incorporate all the plausible properties of a similarity learning method listed in Section 4.1. For instance, if the rules which are used for constructing the RBS and DRBS models are consistent, the resulting similarity function is guaranteed to respect the fundamental feature of a similarity relation in a classification context (see Definition 3.1). Hence, the models are consistent with the training data. For unsupervised RBS this property is difficult to verify in a general case. However, if the semantic concepts which represent the documents are properly assigned (e.g. by experts or a well-trained supervised algorithm), the consistence with data is a natural consequence. Moreover, the similarity function of the RBS model is a proper similarity function if only the data set is consistent and the utilized sets of rules meet a few general requirements (i.e. they are consistent with the data, cover all objects, are minimal and allow to

uniquely identify all objects that originally were discernible – see Section 4.3.3).

By its design RBS takes into consideration the context for evaluation of the similarity. A value of the resulting similarity function depends on a decision class of a referent object. The similarity values are also influenced by a presence of other objects in the data. Due to the utilization of the rough sets (i.e. the use of notions such as a reduct, an uncertainty and membership function, as well as the overall approach which resembles searching for appropriate similarity approximation space), the model is capable of automatically adapting itself to the data at hand. This characteristic contributes to good performance of the RBS models in tasks such as a supervised classification. This fact is confirmed by experiments on real-life data which are described in the next Chapter 5.

The proposed model can be more intuitive for domain experts than typical distance-based approaches. Unlike distance-based metrics, RBS does not enforce any undesirable properties on the induced similarity relation. The set representation, originally borrowed from Tversky's feature contrast model, is more natural for complex objects than the vector representation in a metric space. It is particularly important in situations when the vectors representing objects would have to be high dimensional and possibly sparse (e.g. typical bag-of-words representation of textual documents). The set representation also allows to conveniently model the phenomenon that the lack of some important characteristics in both of compared objects is not an argument for their similarity. Moreover, RBS treats the evaluation of similarity as a problem of resolving conflicts between arguments for and against the similarity, which has an intuitive interpretation.

An important aspect of RBS models is their computational complexity. The construction time of the models depends on particular algorithms used for extracting higher level features. Thanks to the proposed extensions the model can be efficiently built even for very high dimensional data sets. A bigger issue is related to a time cost of a similarity assessment between a single pair of objects. Since the model considers influence of other objects on the context, the computation cost of the RBS similarity function can be in the worst case linear with regard to the number of objects in the data. Since the corresponding cost for typical distance-based similarity functions is constant, such models are easer to apply for analysis on data sets with many objects. On the other hand, the bounded computation cost and robustness with regard to the number of attributes (the sizes of higher-level feature sets can be limited by applying simple filters on rule induction algorithms) makes RBS a useful tool for solving the few-objects-many-attributes problem.

Chapter 5

Experimental Evaluation of the Rule-Based Similarity Model

This chapter presents the results of experiments in which Rule-Based Similarity was used for constructing similarity models from various types of data. The aim of those experiments was to demonstrate feasibility of the rule-based approach to the similarity learning problem. Quality of the proposed model was evaluated using methods briefly described in Section 3.1.4. Depending on the context in which a given similarity model was meant to be applied (i.e. an object classification or a semantic similarity of texts), its quality was judged based on a performance of the 1-nearest neighbour classifier or on a conformity of the similarity function to feedback provided by domain experts.

The performance of the RBS model was additionally compared to several other similarity models as well as to the state-of-the-art classifiers in the investigated domain. For the sake of an in-depth analysis of the results, not only are the raw evaluation values presented but also their statistical significance is given. Although most of those results were already published and presented at respectful conferences [61, 62, 64, 65, 67, 70], some new views at those tests are shown as well. All the experiments described in this chapter were implemented and executed in R System [121]. RBS and its extensions were coded in a native R language with an exception of the discretization algorithm (see Algorithm 2 in Section 4.3.2), which was supported by a C++ code. This code was executed through the *.C* interface provided by R. The whole experimental environment, including the code, data sets and documentation, that allows to conveniently repeat a major part of the conducted experiments is available on request[1].

The chapter is divided into three sections. The first one discusses the performance of the original RBS model. In that section (Section 5.1), RBS was constructed for several benchmark data sets from the UCI repository[2] [36] and compared with several common distance-based models. Next, Section 5.2 shows the evaluation of the proposed model on microarray data sets which are an example of high dimensional data. Finally, in Section 5.3 a case-study of semantic similarity learning from biomedical texts is presented.

[1] *janusza@mimuw.edu.pl*

[2] UC Irvine Machine Learning Repository: http://archive.ics.uci.edu/ml/

5.1 Performance of Rule-Based Similarity in a Classification Context

The original RBS model was tested on a range of benchmark data sets and compared to several commonly used similarity models. Its performance was also verified on a few high dimensional data sets to check its usefulness for learning a similarity relation characterised by multiple possible aspects. This section describes the methodology and presents the results of those test.

5.1.1 Description of the benchmark data sets

The first series of experiments with RBS was conducted on a set of six benchmark data tables, from which five were downloaded from the UC Irvine Machine Learning Repository [36] and one was taken from an R System library *MASS* [121] (the *Cars93* data). They concern domains such as classification of cars, handwritten digits recognition, breast tumour diagnosis and recurrence risk assessment, automatic assessment of nursery applications and Internet advertisements recognition.

A few basic characteristics of the utilized data sets are shown in Table 5.1. They significantly differ in both, the number of objects and attributes. Three of the selected data sets contain nominal attributes, whereas numeric attributes are present in five tables. The *Nursery* data set was the only one containing purely nominal features. The number of decision classes for each set ranged from two (the *WDBC*, *WPBC* and *InternetAds* data sets) to ten (the *Pendigits* data).

Table 5.1: A brief summary of the benchmark datasets used in the experiments with the original RBS model.

Data set:	no. instances	no. attributes	numeric attributes	nominal attributes	no. decision classes
Cars93	93	27	Yes	Yes	6
Pendigits	10992	17	Yes	No	10
WDBC	569	31	Yes	No	2
WPBC	198	33	Yes	No	2
Nursery	12958	9	No	Yes	4
InternetAds	3279	1559	Yes	Yes	2

Additional experiments were performed to assess usefulness of the RBS model for the similarity learning from high dimensional data. For this series of tests, four microarray data sets[3] were selected along with the *InternetAds* data table which was already used in the initial experiments. Two of the chosen microarray data sets (*PTC* and *Barrett*) are smaller benchmark tables, whereas the two other (*HepatitisC* and *SkinPsoriatic*) were obtained from the ArrayExpress repository[4]

[3]For more information on microarray data see Section 5.2.1.

[4]www.ebi.ac.uk/arrayexpress

Table 5.2: A brief summary of microarray data sets used in the experiments.

Data set:	no. samples (instances)	no. genes (attributes)	no. decision classes
PTC	51	16502	2
Barrett	90	22277	3
HepatitisC	124	22277	4
SkinPsoriatic	180	54675	3

[106] and were created as a result of larger research projects (experiment accession numbers E-GEOD-14323 and E-GEOD-13355, respectively).

A microarray analysis is an important source of high dimensional data. A case-based approach to knowledge discovery from collections of microarrays is popular due to scarcity of available data samples [14, 89, 171]. The construction of a reliable similarity model for such a data type is usually a challenging task. Since each data sample is described by numerous attributes (from a few thousands to a few hundred thousands), it is difficult to select those relevant in the considered context. The evaluation of RBS on microarray data was performed to check whether the reduct-based construction of relevant features is effective for high dimensional data.

5.1.2 Compared similarity models

In the experiments, the RBS model was constructed for the classification context (see Section 3.1.2 and Section 4.3.2), which was defined by the decision attributes in the data sets. Several other similarity models were also constructed for each of the decision tables. Some of them, such as the Euclidean distance-based model[5] (the Gower distance-based similarity, see Section 3.2.1), were unsupervised, whereas the others utilized the information about classification of objects to adapt to the data.

Among the supervised similarity models used in this comparison, the most important one was the distance-based model combined with a genetic algorithm for learning parameters of local distances. This model will be called Genetic-Based Similarity (GBS). This approach was described in more details in Section 4.2.2. The genetic algorithm was coded in the native R language [121]. As the local distances it used an absolute difference for numeric attributes and the equivalence test for the nominal ones. The local distance values were aggregated using the Euclidean norm (the Gower metric – see Section 3.2.1).

The value of the parameter that governs the population size (i.e. the number of chromosomes) was set to 1000 for the smaller data sets and to 250 for the larger. The probabilities of the replication, mutation and crossover operations for a particular chromosome were computed using the roulette wheel selection technique, based on a distribution of scores (fitness values) in the population. The exact copies of the chromosomes chosen for replication were taken to the next generation. The chromosomes which were chosen for mutation were randomly modified on a small

[5]For the data sets containing nominal attributes the Gower distance was used.

Table 5.3: A summary of the models used in the experiments from Section 5.1.

Name:	**A short description of a model:**
Gower	A standard Gower distance-based similarity function.
Gower + FS	A Gower distance-based similarity function with a t-statistic filter for selecting relevant attributes.
Minkowski + FS	A Minkowski distance-based similarity function combined with a t-statistic attribute filter and a metric parameter learning wrapper.
GBS	A genetic algorithm-based similarity function learning.
RBS	The original Rule-Based Similarity model.

number of genes (the genes were also chosen at random) and added to the new generation. Next, the chromosomes chosen to crossover were randomly matched in pairs to produce two offspring. The new chromosomes were computed as a weighted averages of the parent chromosomes. Finally, scores of the new generation members were computed and the chromosomes with lower scores were eliminated so that the size of the population did not exceed the starting value. In this way the selection of a chromosome was not directly dependent on its fitness but instead, it was conditioned on its ranking in the population.

Additionally, two different similarity learning models were implemented for the experiments on the high dimensional data sets. Both of those models represented the feature selection approach to similarity learning (see Section 4.2.1). The first one, denoted by Gover+FS, was a combination of the Gower distance-based similarity function with a filter attribute selection method. Relevant features were selected using a t-statistic filter. The attributes were ranked according to average p-values of a t-test (the lower the average, the higher the rank) that check equity of attributes' values within pairs of decision classes. The final number of top-ranked attributes for the model was decided using the leave-one-out cross-validation [18, 80] on the available training data. This number was chosen within the range of 2 to 1000. The second model, called Minkowski+FS, extended the first one by allowing to tune the local distance aggregation function (the p parameter in the Minkowski's aggregation function, see Section 3.2.1). To increase the performance of all the distance-based models, numeric attributes in the data sets were scaled before the experiments.

The RBS model was designed for each of the data sets as it was described in Section 4.3. The relevant higher-level features were constructed from the attributes constituting decision superreducts[6]. The attributes were selected and discretized using a supervised greedy heuristic [71, 97] which was modified so that instead of selecting only one cut at a time, the algorithm was able to simultaneously choose cuts on several attributes that discern most of the samples from different decision classes. The rules which define the higher-level features were discovered using the *decision apriori* algorithm implemented in the *arules* R System library. Only consistent rules[7] were considered with a minimal support factor set to minimum from 5 and 1% of a

[6]A decision superreduct is a set of attributes that discern all objects from different decision classes but does not need to be minimal. See Section 2.3.1.

[7]A rule is called consistent or true if its confidence equals 1. See Section 2.1.3.

total number of objects in a training set. Table 5.3 summarizes the similarity models used in the experiments described in this section.

5.1.3 Evaluation method and discussion of the results

The quality of the compared similarity models was evaluated indirectly by measuring classification performance of 1-NN classification rule (Definition 3.1) applied to the corresponding similarity functions. This similarity model evaluation method was discussed in Section 3.1.4. The classification accuracy (ACC), defined as:

$$ACC = \frac{|\{u \in TestSet : \hat{d}(u) = d(u)\}|}{|TestSet|}, \tag{5.1}$$

where $TestSet$ is a set of test objects and $\hat{d}(u)$ is a prediction of a decision class for an object u, was estimated using the 10-fold cross-validation technique [27]. The cross-validation was repeated 12 times with different partitioning of data sets into folds. Although in each cross-validation run the division of data was random, the same partitioning was used for every tested similarity model in order to facilitate the comparison of the evaluation results. The mean and standard deviations of model accuracies were computed and the significance of differences in results was assessed using the paired t-test with a 0.99 confidence level.

Table 5.5 shows the mean and standard deviation of accuracy obtained by similarity models described in Section 5.1.2 for the regular data sets. Figure 5.1 also conveniently visualizes those results.

Table 5.4: A comparison of the classification accuracy (ACC) of the tested models.

Dataset:	Gower acc. (%)	GBS acc. (%)	RBS acc. (%)
Cars93	63.44 ± 2.41	87.96 ± 1.11	89.25 ± 1.10
Pendigits	97.46 ± 0.21	98.57 ± 0.26	97.30 ± 0.55
WDBC	95.20 ± 0.31	95.66 ± 0.64	95.53 ± 0.48
WPBC	73.13 ± 1.25	76.25 ± 0.82	76.79 ± 0.85
Nursery	76.28 ± 0.39	78.35 ± 0.31	97.02 ± 0.05
InternetAds	96.52 ± 0.09	96.06 ± 0.44	96.07 ± 0.14

The classification accuracies of the similarity models on the benchmark data are comparable. The RBS model achieved significantly better results on the data sets containing nominal attributes, with an exception of the *InternetAds* data. Although the accuracy of the RBS model for the most of datasets was slightly higher than the accuracy of the GBS model, the difference was significant (p-value of a t-test was lower than 0.01) only for *Cars93*, *Nursery* in favour of the RBS and *Pendigits* in favour of the GBS. However, it is worth noticing that the time needed to perform the tests was much shorter for the rule-based approach.

Comparing to the simple Gover distance-based approach, RBS turned out to be more reliable for all data tables, except *Pendigits* and *InternetAds*. The average classification accuracy of RBS was statistically higher (p-value of a t-test ≤ 0.01)

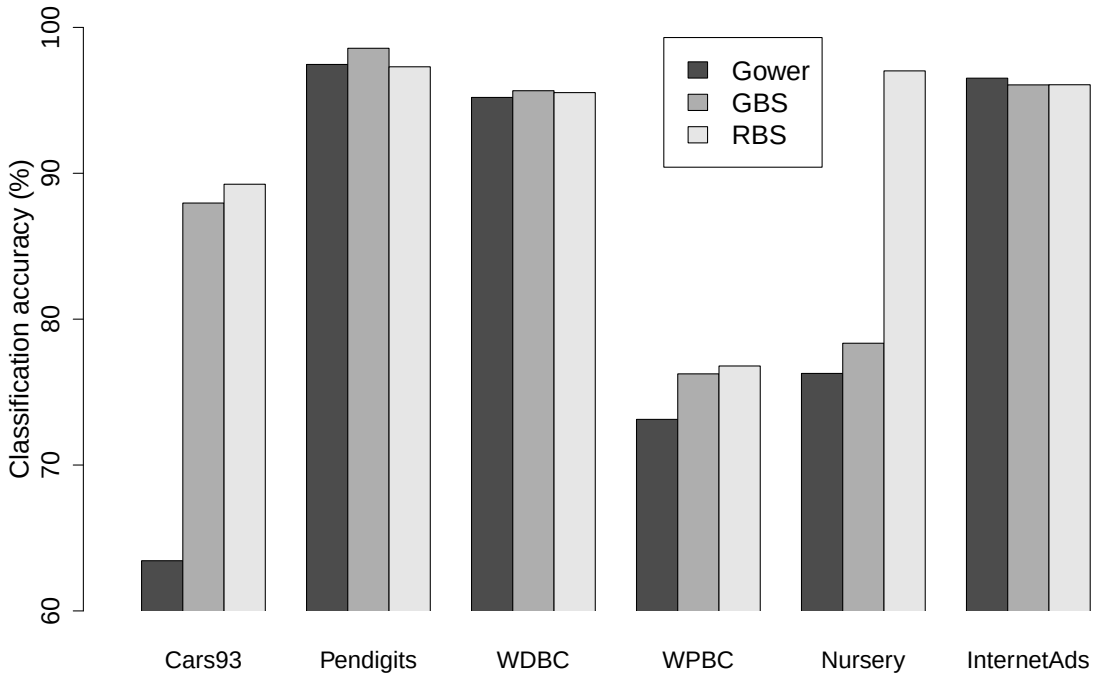


Figure 5.1: A visualization of classification accuracy obtained by the compared similarity models on the benchmark data sets.

for the *Cars93*, *WPBC* and *Nursery* data. Interestingly, the performance of RBS was significantly lower than the performance of the Gower model for the *InternetAds* data set. This fact can be treated as an argument for a hypothesis that the RBS model may fail to capture all relevant aspects of similarity when the dimensionality of a data set is high.

To further investigate this problem the second series of experiments was conducted, in which the performance of RBS was compared to several distance-based models on high dimensional data sets. Table 5.4 shows the results of those tests. They are also displayed in Figure 5.2.

Table 5.5: A comparison of the classification accuracy (ACC) of several similarity models for high dimensional data sets.

Dataset:	Gower	Gower+FS	Minkowski+FS	GBS	RBS
InternetAds	96.52±0.09	96.79±0.14	96.75±0.12	96.06±0.44	96.07±0.14
PTC	84.31±1.41	96.08±1.67	98.04±0.77	95.74±1.95	98.04±1.31
Barrett	51.67±2.23	55.11±1.86	59.78±1.43	55.55±1.97	62.56±2.12
HepatitisC	86.36±1.66	84.54±1.58	85.08±1.06	84.83±1.38	86.58±0.83
SkinPsoriatic	71.17±1.50	70.83±2.13	69.50±2.39	72.17±1.56	79.00±1.12

The results seem to confirm a hypothesis that similarity learning may have a significant impact on a quality of a similarity model for high dimensional data. For

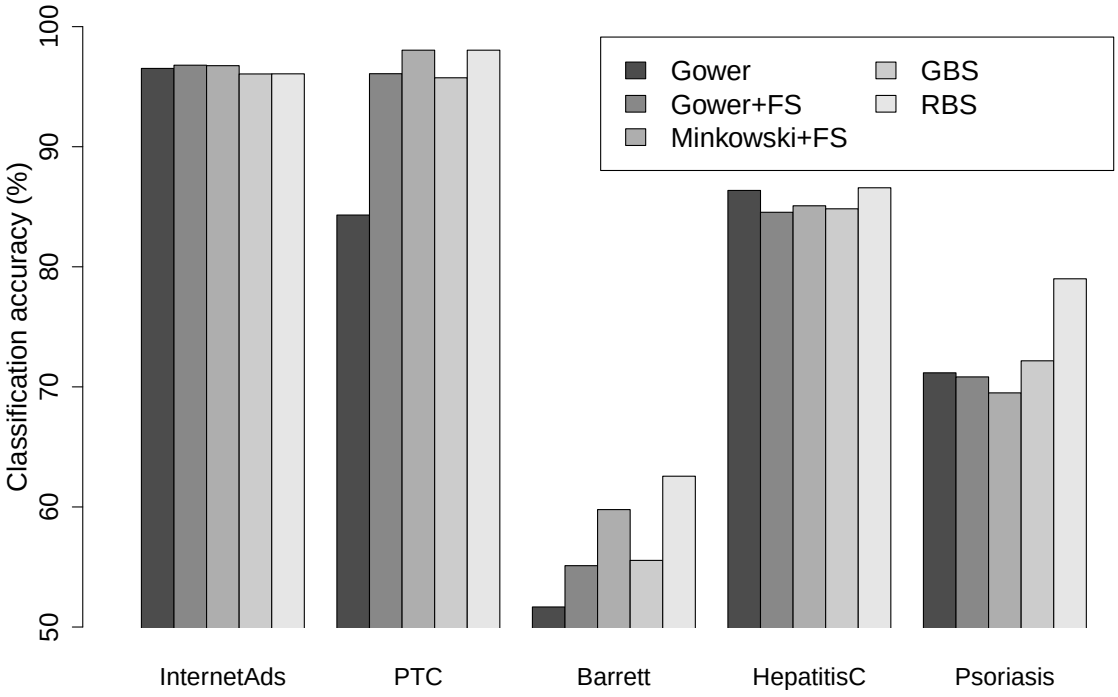


Figure 5.2: A visualization of classification accuracy obtained for the high dimensional data sets.

the *PTC* and *Barrett* data the basic Gower distance-based model, which does not adapt to particular data sets, achieved much lower accuracies than all other similarity models. Moreover, the accuracy of the Gover model was lower than the accuracy of RBS on every data table except *InternetAds* and the difference was statistically significant for the *PTC*, *Barrett* and *SkinPsoriatic* data. On the other hand, its results on the *InternetAds* and *HepatitisC* data sets show that even such a simple model may be sufficient to obtain comparable, if not better, results to much more sophisticated approaches, like the genetic algorithm-based similarity learning (GBS) or RBS.

Accuracy scores achieved by RBS were significantly higher (p-value < 0.01) than those of other similarity learning models for the *Barrett*, *HepatitisC* and *SkinPsoriatic* data. In particular, on average RBS turned out to be more reliable than GBS for all data sets except *InternetAds*. The genetic approach does not work very well for high dimensional data. Its probably due to the over-fitting problem which is likely to happen when extensive supervised tuning is performed for models with many parameters [93]. It has been observed, however, that for data sets with many potentially important attributes (i.e. *InternetAds*, *HepatitisC*) the results of RBS are comparable to those of the much simpler models (Gower, Gower+FS). This, perhaps, can be explained by the fact that RBS was using a much lower number attributes than the other models (the total number of attributes used by RBS for a single data set never exceeded 70, whereas for other models it often was more than ten times greater). On one hand, this characteristic can be advantageous since it facilitates interpretability of the model. On the other hand, however, it may deteriorate the

performance of the model for complex classification problems, when the number of important features is usually high.

5.2 Evaluation of the Dynamic Rule-Based Similarity Model on Microarray Data

The construction of a similarity model for a high dimensional data may require incorporation of numerous characteristics or factors that have an impact on similarity judgements in a given context. The DRBS model was proposed in order to enable working on multiple features during the construction of the model, while keeping the reliability and flexibility of RBS which are provided by utilization of notions from the theory of rough sets. This section shows some applications of DRBS to analysis of several microarray data sets. It also presents the comparison of classification results of the 1-NN algorithm which uses a DRBS-induced similarity function, and a few state-of-the-art classifiers that are commonly employed for microarray data.

5.2.1 Microarrays as an Example of Real-Life High Dimensional Data

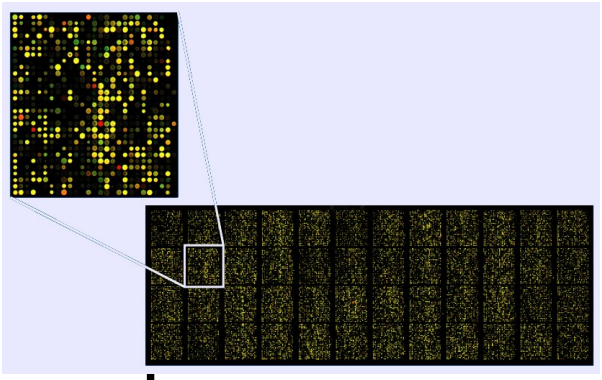
The microarray technology allows researchers to simultaneously monitor thousands of genes in a single experiment. In a microarray data set, specific microarray experiments are treated as objects (e.g. tissue samples). The attributes of those objects correspond to different genes and their values correspond to *expression levels* – the intensity of a process in which information coded in a gene is transformed into a specific gene product. Figure 5.3 visualizes a single microarray chip after an experiment and its representation in a decision table.

In recent years, a lot of attention of researchers has been put into investigation of this kind of data. That growing interest is largely motivated by numerous practical applications of knowledge acquired from microarray analysis in medical diagnostics, treatment planning, drugs development and many more [3]. When dealing with microarrays, researchers have to overcome the problem of insufficient availability of data. Due to very high costs of microarray processing, usually the number of examples in data sets is limited to several dozens. This fact, combined with a large number of examined genes, makes many of the classic statistical or machine learning models unreliable and encourages researchers to develop specialized methods for solving the *few-objects-many-attributes* problem [139].

Thorough experiments have been conducted to test the performance of the DRBS model on 11 microarray data sets. All the data samples were downloaded from the ArrayExpress[8] repository. All the data available in the repository are in the MIAME[9] standard [19]. To find out more about this open repository refer to [106]. Each of the used data sets was available in a partially processed form as two separate files. The first one was a data table which contained information about expression levels of genes

[8] www.ebi.ac.uk/arrayexpress

[9] **M**inimal **I**nformation **A**bout **M**icroarray **E**xperiment



Microarray data:
few-objects-many-attributes problem

≈40k genes (attributes)

sampleID	AFFX-3_at	3322_i_at	4969_s_at	...	22095_s_at	22379_at	Diagnosis
GSM1.CEL	4.010	12.434	32.443	...	1.665	12.434	3
GSM2.CEL	5.314	43.765	5.763	...	3.567	7.645	2
GSM3.CEL	3.275	17.567	23.842	...	0.657	12.446	2
GSM4.CEL	2.112	8.432	54.849	...	87.656	45.324	1
...	...	...	...	...	...	...	...
GSM149.CEL	8.453	10.087	8.678	...	2.986	9.656	3

Figure 5.3: A visualization of a microarray chip after an experiment (the top left corner) and its representation in a decision system. The intensity of a colour of spots at the chip reflects expression levels of the genes.

in particular samples and the second was a *SDRF*[10] file storing meta-data associated with samples (e.g. decision classes). Entries in those files had to be matched during the preprocessing phase. Figure 5.4 shows a standard microarray data preprocessing schema.

The data sets used in experiments were related to different medical domains and diverse research problems (the ArrayExpress experiment accession numbers are given in parentheses):

1. *AcuteLymphoblasticLeukemia* (ALL) – the recognition of acute lymphoblastic leukemia genetic subtypes (E-GEOD-13425).
2. *AnthracyclineTaxaneChemotherapy* (ATC) – the prediction of response to anthracycline/ taxane chemotherapy (E-GEOD-6861).
3. *BrainTumour* (BTu) – diagnosis of human gliomas (E-GEOD-4290).
4. *BurkittLymphoma* (BLy) – the diagnostics of human Burkitts lymphomas (E-GEOD-4475).
5. *GingivalPeriodontits* (GPe) – transcription profiling of human healthy and diseased gingival tissues (E-GEOD-10334).
6. *HeartFailurFactors* (HFF) – transcription profiling of human heart samples with different failure reasons (E-GEOD-5406).

[10] **S**ample and **D**ata **R**elationship **F**ile

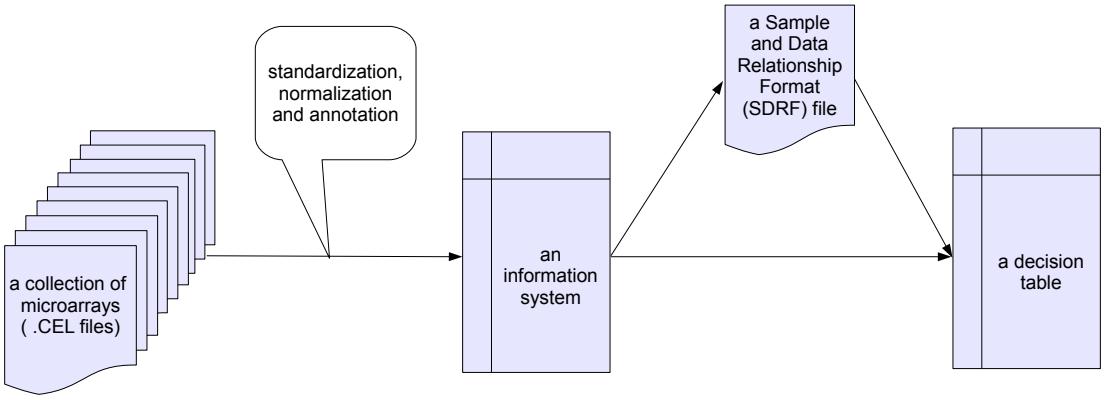


Figure 5.4: A standard preprocessing schema for microarray data sets.

7. *HepatitisC* (HeC) – an investigation of a role of the chronic hepatitis C virus in the pathogenesis of HCV-associated hepatocellular carcinoma (E-GEOD-14323).

8. *HumanGlioma* (HGl) – the recognition of genomic alterations that underlie brain cancers (E-GEOD-9635).

9. *OvarianTumour* (OTu) – the recognition of the ovarian tumour genetic subtypes (E-GEOD-9891).

10. *SepticShock* (SSh) – profiling of critically ill children with the systemic inflammatory response syndrome (SIRS), sepsis, and septic shock spectrum (E-GEOD-13904).

11. *SkinPsoriatic* (SPs) – an investigation of genetic changes related to the skin psoriasis (E-GEOD-13355).

Apart from matching the decisions to samples some additional preprocessing was needed to remove those decision classes which were represented by less than 3 instances. The first 10 data sets were previously used in RSCTC'2010 Discovery Challenge [168]. The eleventh set was previously used for the comparison of the original RBS with distance-based similarity learning models in [64] (see Section 5.1). A part of those data sets was also used in the preliminary experiments, in which a developing version DRBS was compared to the original RBS model ([65]). Table 5.6 presents some basic characteristics of the data sets. They differ in the number of samples (from 124 to 284), the number of examined genes (it varies between 22276 and 61358) and decision classes (2 to 5). Only data sets which contained more than 100 samples were used in the experiments with DRBS.

Some of the data sets have significantly uneven class distribution, with one dominant class represented by majority of samples and a few minority classes represented by a small number of objects. Typically, in microarray data, the minority

Table 5.6: A brief summary of microarray data sets used in the experiments.

Data set:	no. samples	no. genes	no. classes (& class distribution)
ALL	190	22276	5 (0.28, 0.23, 0.19, 0.23, 0.07)
ATC	160	61358	2 (0.59, 0.41)
BTu	180	54612	4 (0.28, 0.13, 0.14, 0.45)
BLy	221	22282	3 (0.20, 0.58, 0.22)
GPe	247	54674	2 (0.74, 0.26)
HFF	210	22282	3 (0.51, 0.41, 0.08)
HeC	124	22276	4 (0.14, 0.38, 0.15, 0.33)
HGl	186	59003	5 (0.57, 0.18, 0.08, 0.07, 0.10)
OTu	284	54620	3 (0.87, 0.06, 0.07)
SSh	227	54674	5 (0.47, 0.23, 0.12, 0.08, 0.10)
SPs	180	54675	3 (0.32, 0.36, 0.32)

classes are more interesting than the dominant one and this fact should be reflected by the quality measure used to assess the performance of classification algorithms. For this reason, the quality of the models employed in the experiments was evaluated using the *balanced accuracy* (BAC) measure. This is a modification of the standard classification accuracy (Eq. 5.1) which is insensitive to imbalanced frequencies of decision classes. It is calculated by computing standard classification accuracies ($Accuracy_i$) for each decision class and then averaging the result over all classes ($d = 1, \ldots, l$). In this way, every class has the same contribution to the final result, no matter how frequent it is:

$$\begin{aligned} ACC_i &= \frac{|\{u \in TestSet : \hat{d}(u) = d(u) = i\}|}{|\{u \in TestSet : d(u) = i\}|}, \\ BAC &= \Big(\sum_{i=1}^{l} ACC_i\Big)/l \quad , \end{aligned} \tag{5.2}$$

where l is a total number of decision classes, $TestSet$ is a set of test samples and $\hat{d}(u)$ is a prediction for a sample u. In a case of a 2-class problem with no adjustable decision threshold, balanced accuracy is equivalent to *Area Under the ROC Curve* (AUC). Thus, it may be viewed as a generalization of AUC for multi-class classification problems. Balanced accuracy is insensitive to imbalanced class distribution. This particular measure was used during RSCTC'2010 Discovery Challenge [168] to evaluate solutions of participants and it is also used in the experiments described further in this section.

5.2.2 Comparison with the state-of-the-art in the microarray data classification

The performance of DRBS for microarray data sets was evaluated in two series of experiments. In the first one, DRBS was compared to the original RBS model and

three distance-based approaches. The distance-based models used different feature selection techniques combined with a Minkowski distance-based similarity measure (see Section 4.2.1) whose parameter p was automatically tuned on available training data. The utilized feature selection methods were based on *correlation test* [52], *t-test* [87, 91] and the *relief* algorithm [79], respectively. Table 5.7 shows the results of this comparison for six data tables from the basic track of RSCTC'2010 Discovery Challenge [168]. The results are also visualized in Figure 5.5.

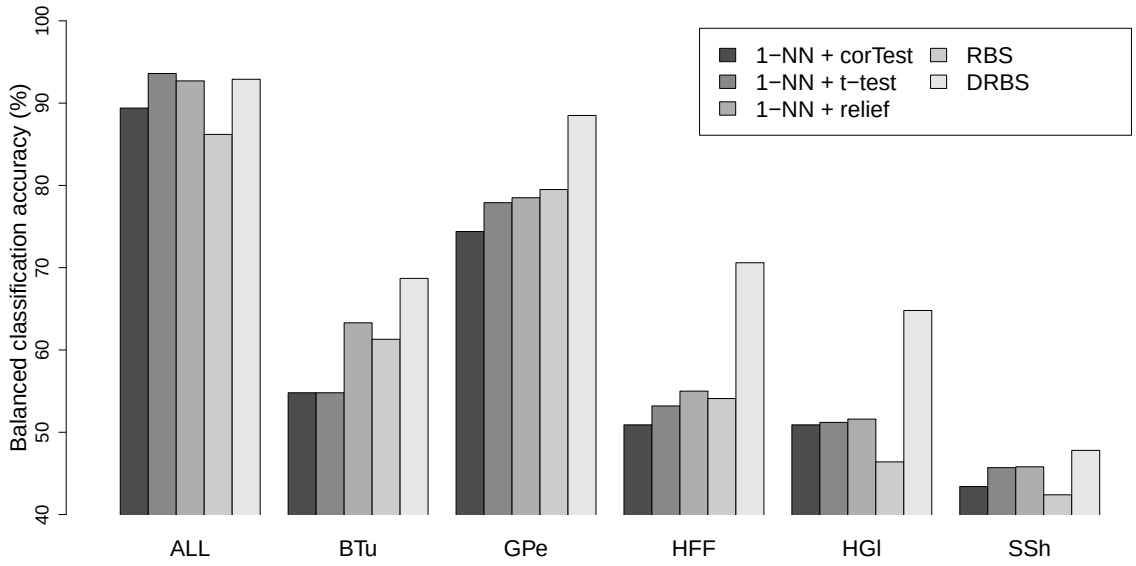


Figure 5.5: Balanced classification accuracies of the compared similarity models.

Table 5.7: Results of different similarity models for microarray data sets. For each table, the best score is marked in red and the second best is in blue. Mean and standard deviation values are given.

Data set:	1-NN+$corTest$	1-NN+$tTest$	1-NN+$relief$	RBS	DRBS
ALL	89.4±2.4	93.6±2.3	92.7±1.7	86.2±1.7	92.9±0.8
BTu	54.8±1.0	54.8±2.8	63.3±2.1	61.3±2.7	68.7±1.0
GPe	74.4±1.9	77.9±1.6	78.5±2.5	79.5±1.8	88.5±1.6
HFF	50.9±2.3	53.2±2.9	55.0±1.9	54.1±1.1	70.6±2.2
HGl	50.9±2.3	51.2±3.3	51.6±1.8	46.4±1.9	64.8±1.3
SSh	43.4±3.2	45.7±2.2	45.8±2.4	42.4±2.3	47.8±1.7
avg. BAC	60.6±17.5	62.7±18.8	64.5±17.9	61.7±17.8	72.2±16.5

From the results of those experiments it is clearly visible that DRBS is a viable improvement over RBS for high dimensional microarray data. Not only did DRBS achieve better balanced accuracy than RBS for each of the data sets but also the differences in their results were always statistically significant. Comparing to the distance-based models, DRBS performed better for five out of six data sets. Only for

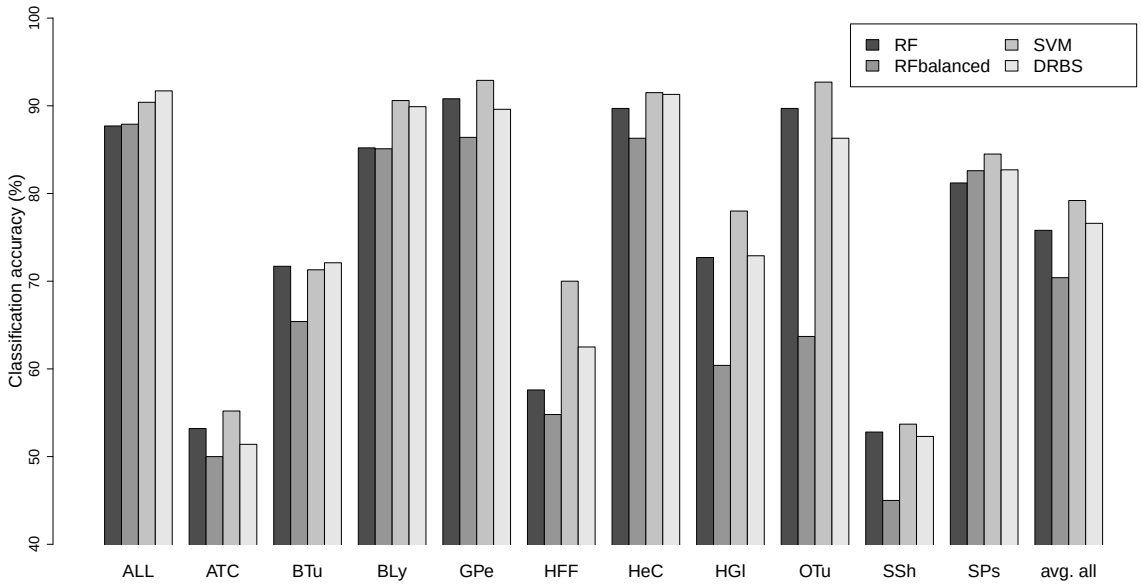


Figure 5.6: A visualization of the classification accuracies obtained by the compared algorithms.

the *AcuteLymphoblasticLeukemia* data (ALL) the average BAC score of 1-NN model combined with the t-test filter turned out to be higher, yet even in that case, the difference could not be marked as statistically significant. Unexpectedly, the most reliable gene selection method for the distance-based models was the *relief* algorithm, which was ranked second for four tables.

In the second series of experiments, the classification performance achieved with a combination of DRBS and the simple classification rule from Definition 3.1 was compared to the results of the Random Forest [20, 29] and SVM [37, 87] algorithms which are considered as the state-of-the-art in the biomedical data domain. All the models were implemented in R System ([121]). The DRBS model consisted of $(0.9, 0.95)$–dynamic reducts (see Definition 2.12) constructed from 250 randomly selected subsets of $5 * \lfloor\sqrt{|A|}\rfloor$ genes, were $|A|$ is a total number of genes (attributes) in a data set. These values guaranteed that a probability of an inclusion of any particular gene to at least one random subset was greater than 0.95 (see Section 4.3.4). Those particular parameter values were chosen as a trade off between computational requirements and robustness of the model. No tuning of the parameters was performed during the experiments due to computation complexity reasons, but it was observed that, for several different data sets, an increase in the number of random subsets of genes usually leads to a slightly better classification quality.

Apriori algorithm from the *arules* package was used for the generation of the rule sets for DRBS. The implementation of Random Forest from the package *randomForest* was used with parameter settings recommended in [29]. Additionally, a balanced version of RF model was checked in which empirical probabilities of decision classes (computed on a training set) were used during the voting as a cut-off values. Support Vector Machine was trained with a linear kernel. The implementation from the package *e1071* was used. Other parameters of SVM were set to values used by the

Table 5.8: Results of the tests evaluated using the classification accuracy (ACC) measure. For each data set, the best score is marked in red and the second best is in blue. Mean and standard deviation values are given.

Data set:	RF	RF*balanced*	SVM	DRBS
ALL	87.98 ± 0.97	88.77 ± 1.19	90.39 ± 0.96	91.71 ± 0.60
ATC	53.28 ± 2.90	49.64 ± 3.72	55.73 ± 2.78	51.35 ± 3.45
BTu	71.30 ± 1.26	66.44 ± 1.68	71.44 ± 1.45	72.08 ± 1.16
BLy	86.01 ± 1.65	86.05 ± 1.17	90.54 ± 1.79	89.89 ± 1.74
GPe	90.69 ± 1.02	86.50 ± 0.55	92.95 ± 0.90	89.57 ± 1.35
HFF	59.29 ± 1.75	56.03 ± 2.35	70.28 ± 1.81	62.54 ± 2.45
HeC	89.92 ± 1.52	87.16 ± 1.40	91.60 ± 1.80	91.26 ± 1.57
HGl	72.45 ± 1.91	61.74 ± 2.11	77.96 ± 1.23	72.76 ± 1.13
OTu	89.61 ± 0.41	64.91 ± 1.72	92.66 ± 0.52	86.27 ± 1.07
SSh	52.57 ± 1.53	44.49 ± 3.24	53.71 ± 2.48	52.31 ± 1.41
SPs	81.16 ± 1.47	82.64 ± 0.82	84.77 ± 1.45	82.69 ± 1.29
avg. ACC	75.84 ± 14.98	70.40 ± 16.44	79.27 ± 14.67	76.58 ± 15.42

winners of the advanced track of RSCTC'2010 Discovery Challenge [168]. No gene selection method was used for any of the compared models.

The quality of the compared models was assessed using two different quality measures – mean accuracy (ACC, Eq. 5.1) and balanced accuracy (BAC, Eq. 5.2). Those measures highlight different properties of a classification model. By its definition, the balanced accuracy gives more weight to instances from minority classes, whereas the standard mean accuracy treats all objects alike and, as a consequence, usually favours the majority class. Depending on applications, each of those properties can be useful, thus, a robust classification model should be able to achieve a high score regardless of the quality measure used for the assessment. The tests were performed using 5-fold cross validation technique. The experiments were repeated 12 times for each of the data sets and models. This testing methodology has been proved to yield reliable error estimates in terms of bias and standard deviation (see [18, 27, 80]). The results in terms of the accuracy and the balanced accuracy are given in Tables 5.8 and 5.9, respectively.

As expected, there were significant differences between performances of the models depending on the quality measure used for the assessment. In terms of the accuracy, SVM turned out to be the most reliable. It achieved the best score on 9 data sets, whereas DRBS scored the best on 2 data tables. Different results were noted in terms of the balanced accuracy – DRBS and Random Forest (the balanced version) had the highest mean score on 4 sets, whereas SVM ranked first on 3 data sets. Pairwise comparisons of the tested models are summarized in Tables 5.10 and 5.11. For each pair, a number of data sets for which the model named in a column achieved a higher average score is given.

The statistical significance of differences in the results between each of models was verified using the paired Wilcoxon test. This particular statistical test was used

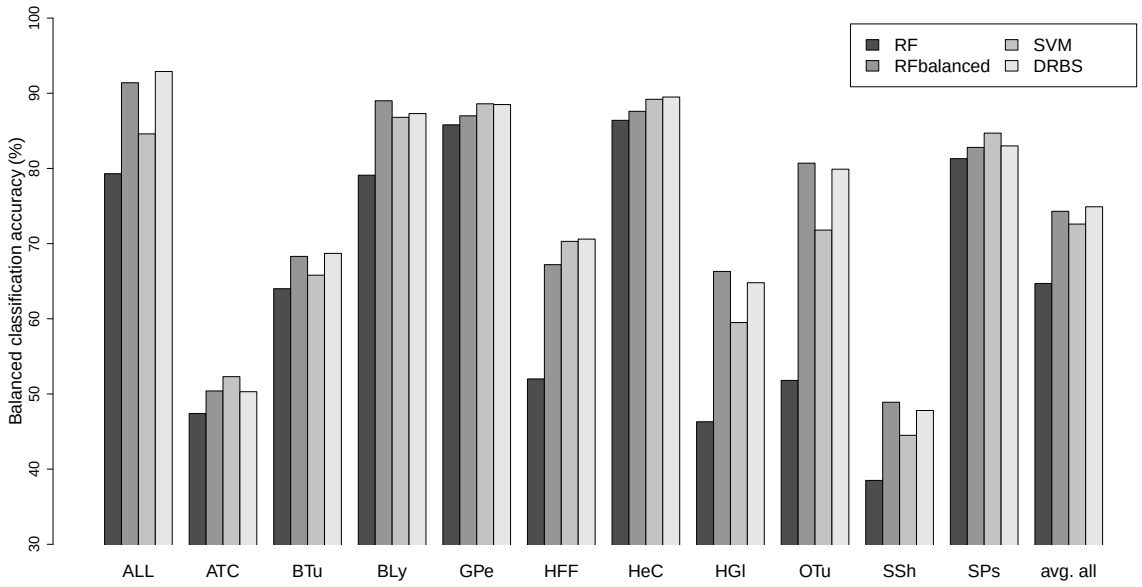


Figure 5.7: A visualization of the balanced classification accuracies obtained by the compared algorithms.

Table 5.9: Results of the tests evaluated using the balanced accuracy (BAC) measure. For each data set, the best score is marked in red and the second best is in blue. Mean and standard deviation values are given.

Data set:	RF	RF*balanced*	SVM	DRBS
ALL	79.34 ± 2.08	91.40 ± 1.25	84.68 ± 2.68	92.93 ± 0.77
ATC	47.28 ± 2.92	50.46 ± 4.08	52.92 ± 2.97	50.33 ± 3.39
BTu	63.93 ± 1.80	68.49 ± 1.83	65.88 ± 1.82	68.71 ± 1.04
BLy	79.15 ± 2.30	89.08 ± 1.32	86.65 ± 2.52	87.30 ± 2.17
GPe	85.88 ± 1.50	87.04 ± 0.70	88.76 ± 1.30	88.52 ± 1.59
HFF	51.98 ± 3.13	67.17 ± 1.72	70.62 ± 2.10	70.64 ± 2.18
HeC	86.42 ± 1.78	87.55 ± 1.36	89.28 ± 1.57	89.52 ± 1.55
HGl	46.35 ± 2.83	66.46 ± 1.43	59.49 ± 2.10	64.76 ± 1.32
OTu	51.79 ± 1.67	80.75 ± 2.23	71.8 ± 1.80	79.91 ± 2.25
SSh	38.08 ± 1.98	48.98 ± 2.94	44.51 ± 2.13	47.77 ± 1.68
SPs	81.32 ± 1.43	82.80 ± 0.80	84.87 ± 1.42	82.95 ± 1.29
avg. BAC	64.68 ± 18.16	74.56 ± 15.22	72.68 ± 15.59	74.85 ± 15.71

instead of the standard t-test because balanced accuracies of different classifiers are not likely to have a normal distribution with equal variances. A null hypothesis was tested that the true performance measurements obtained for the particular data set have equal means. Due to a large number of the required comparisons a Bonferroni correction was applied and each test was conducted on 0.9999 confidence level. Differences in means were marked as significant (i.e. the null hypothesis was

Table 5.10: A pairwise comparison of accuracies (ACC) of the tested models. Tables show the number of data sets for which the model named in a column achieved a higher score. The number of statistically significant wins is given in parentheses.

Method name: lower\higher	RF (higher)	RF*balanced* (higher)	SVM (higher)	DRBS (higher)
RF (lower)	–	3 (1)	11 (9)	7 (4)
RF*balanced* (lower)	8 (8)	–	11 (11)	11 (9)
SVM (lower)	0 (0)	0 (0)	–	2 (1)
DRBS (lower)	4 (1)	0 (0)	9 (6)	–

rejected and a statistical proof was found that performance of one of the model is higher) if the p-value[11] of the test was lower than 0.01. The results of this comparison are also shown in Tables 5.10 and 5.11 (in parentheses).

It is worth noticing that DRBS turned out to be the most stable classification model – differences in its score in terms of the accuracy and the balanced accuracy were the smallest of the tested models. For example, although SVM achieved the highest average accuracy on all data sets (79.27), the average difference between its accuracy and the balanced accuracy was 6.59. The value of the same statistic for DRBS was 1.73, with average accuracy of 76.58 (it ranked second in terms of the accuracy measure). DRBS achieved the highest average balanced accuracy of 74.85. This score was only slightly higher than the result of the second algorithm – balanced Random Forest (74.56). The results of the Random Forest algorithms, however, significantly differed with regard to the quality measures. The absolute differences between average values of the two utilized indicators for the Random Forest and balanced Random Forest models were 11.16 and 4.16, respectively. Those results clearly show that DRBS can successfully compete with the state-of-the-art in the microarray data classification.

Table 5.11: A pairwise comparison of balanced accuracies (BAC) of the models. Tables show the number of data sets for which the model in a column achieved a higher score. The number of statistically significant wins is given in parentheses.

Method name: lower\higher	RF (higher)	RF*balanced* (higher)	SVM (higher)	DRBS (higher)
RF (lower)	–	11 (8)	11 (10)	11 (10)
RF*balanced* (lower)	0 (0)	–	5 (3)	6 (3)
SVM (lower)	0 (0)	6 (6)	–	8 (5)
DRBS (lower)	0 (0)	5 (2)	3 (1)	–

[11]The p-value of a statistical test is the probability of obtaining a test statistic value as extreme as the observed one, assuming that the null hypothesis of the test is true.

5.3 Unsupervised Similarity Learning from Textual Data

This section demonstrates an application of the unsupervised RBS model for computation of semantic similarity of texts. Reliable semantic similarity assessment is crucial for numerous practical problems, such as information retrieval [54, 122], clustering of documents or search results [55, 95], or multi-label classification of textual data [68]. The usefulness of unsupervised RBS in one of those tasks, namely document grouping, is verified on a corpus of scientific articles related to biomedicine. The notion of information bireducts (see Section 2.3.3) is combined with Explicit Semantic Analysis (ESA) (see Section 4.2.4) in order to extract important features of the texts, and the performance of unsupervised RBS is compared to the cosine similarity model.

5.3.1 Testing Methodology

The experiments were performed on a document corpus consisting of 1000 research papers related to biomedicine which were downloaded from PubMed Central repository [123]. The ESA algorithm, which was used for extracting semantic features of the texts, was adapted to work with the MeSH ontology [161] and implemented in R System [121]. Prior to the experiments, documents from the corpus were processed with natural language processing tools from the *tm* and *RStem* libraries, and the associations between the documents and the MeSH headings were precomputed[12]. The documents were represented by *bags-of-concepts* (see Section 4.2.4) to construct the unsupervised RBS model described in Section 4.3.5, as well as the other models used for comparison (see Section 5.3.2).

Additionally, all of the documents were manually labelled by experts from the U.S. National Library of Medicine (NLM) with the MeSH subheadings [161]. Those labels represent a topical content of the documents and as such, they can serve as means for evaluation of truly semantic relatedness of the texts (see Section 3.1.4). In the presented experiments, they were used for computation of a semantic proximity between the analysed documents, which is treated as a reference for the compared similarity functions.

The semantic proximity was measured using F_1-*distance*, defined as:

$$F_1\text{-}distance(T_i, T_j) = 1 - 2 \cdot \frac{precision(S_i, S_j) \cdot recall(S_i, S_j)}{precision(S_i, S_j) + recall(S_i, S_j)}, \tag{5.3}$$

where S_i and S_j are sets of labels (MeSH subheadings) assigned by experts, that represent documents $T_i, T_j \in D$, respectively, and

$$precision(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_i|}, \qquad recall(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_j|}.$$

[12]The corpus used in the experiments is a subset of a data set prepared for JRS'2012 Data Mining Competition [68, 72]. For more details on the contest and data preprocessing refer to http://tunedit.org/challenge/JRS12Contest)

This particular measure was chosen since it is often used for evaluation of results in the information retrieval setting [35, 78]. Although the evaluation of a similarity measure by a distance metric may fail to capture some of the psychologically important properties and underestimate its real quality, in this way it was possible to quantitatively assess many similarity models and use those assessments to objectively compare them (see Section 3.1.4).

The semantic proximity between two sets of documents is defined as an average of F_1-*distance* between each pair of texts from different sets:

$$semDist(D_1, D_2) = \frac{\sum_{T_i \in D_1, T_j \in D_2} F_1\text{-}distance(T_i, T_j)}{|D_1| \cdot |D_2|} \ . \tag{5.4}$$

In experiments, three types of comparisons between the similarity models were made. In the first one, for each of the models, its correlation with the semantic proximity values (Eg. 5.3) was computed. This kind of evaluation is often used in psychological studies where researchers try to measure the dependence between values returned by their models and assessments made by human subjects [42, 148, 159]. It is also commonly utilized in studies on semantic similarity of texts [38].

One disadvantage of this evaluation method is that the linear correlation does not necessarily indicate the usefulness of the measure in practical applications such as the information retrieval or clustering. For this reason, the second series of tests was performed, which aimed at measuring semantic homogeneity of clusterings resulting from the use of different similarity models. To each document in the corpus there was assigned its average semantic proximity (Eg. 5.4) to other documents from the same cluster and to the remaining texts. If for a division of data into k groups we denote documents from a corpus D belonging to the same cluster as T_i by $cluster_{T_i}$, then we can define a semantic homogeneity of T_i with regard to the semantic proximity function *semDist* as:

$$\begin{aligned} homogeneity(T_i) &= \frac{B(T_i) - A(T_i)}{\max\big(A(T_i), B(T_i)\big)}, && \text{where} \\ A(T_i) &= semDist(T_i, cluster_{T_i} \setminus T_i) && \text{and} \\ B(T_i) &= semDist(T_i, D \setminus cluster_{T_i}). \end{aligned}$$

If $cluster_{T_i} \setminus T_i = \emptyset$, then it is assumed that $homogeneity(T_i) = 1$. The average semantic homogeneity of all documents can be used as a measure of clustering quality. Since useful similarity models should lead to meaningful clustering results, the average semantic homogeneity can be employed to intuitively evaluate the usefulness of the compared similarity models for the clustering task.

Finally, in the last series of tests, it was measured how clustering separability is influenced by different similarity models. Two hierarchical clustering algorithms, agnes (AGglomerative NESting) and diana (DIvisive ANAlysis), were used in the experiments. They are described in [78]. Those algorithms differ in the way they form a hierarchy of data groups. Agnes starts by assigning each observation to a different (singleton) group. In the consecutive steps, the two *nearest* clusters are combined to form one larger cluster. This operation is repeated until there remains only a single cluster. The distance between two clusters is evaluated using a linkage function (see

the brief discussion in Section 3.3.3). To maximize the semantic homogeneity of the clusters, in the experiments the *complete linkage* method was used.

The diana algorithm starts with a single cluster containing all observations. Next, the clusters are successively divided until each cluster contains a single observation. At each step, only a single group, with the highest internal dissimilarity is split. Two different algorithms were used in the experiments to verify the stability of the compared similarity models and avoid the bias towards a single clustering method.

Apart from a clustering hierarchy, those algorithms return agglomerative (AC) and divisive coefficients (DC), respectively. These coefficients express conspicuousness of a clustering structure in a clustering tree [78]. Although they are internal measures and their value does not necessarily correspond to the semantic relatedness of objects within the clusters, they can give an intuition on interpretability of a clustering solution.

5.3.2 Compared Similarity Models

Four similarity models were implemented in R System [121] for the purpose of the experiments. The unsupervised RBS was constructed as described in Section 4.3.5. The documents from the corpus were given associations to MeSH headings using ESA. An information system $\mathbb{S} = (D, F)$ was constructed consisting of 1000 documents described by a total of $25,640$ semantic features. During preprocessing, the features which were not present in at least one document from the corpus D were filtered out from further analysis. Numerical association values of each term were transformed into four distinct symbolic values. The discretization was based on general knowledge of the data (e.g. for each feature possible association values ranged from 0 to 1000) and the cut thresholds were constant for every feature (i.e. they were set to $= 0$, ≥ 300, ≥ 700 and ≥ 1000).

From the discretized information system, 500 information bireducts (see Section 2.3.3) were computed using random permutations (see Algorithm 4). As expected, they significantly differ in selection of features and reference documents. On average, a bireduct consisted of 210 attributes (min = 173, max = 246), with each attribute belonging on average to 9 bireducts (min = 1, max = 42). The average number of documents in a single bireduct was 995 (min = 988, max = $1,000$), and each document appeared on average in 498 bireducts (min = 489, max = 500). All of the computed information bireducts were used for assessment of similarity by the unsupervised RBS model.

Apart from the unsupervised RBS, for the sake of comparison three other similarity models were implemented. The first one was the standard cosine similarity. In this model, for documents T_i and T_j, represented by vectors C_i, C_j of numerical association strengths to headings from the MeSH ontology (i.e. the vector representation of *bag-of-concepts* described in Section 4.2.4), the cosine similarity is:

$$Sim_{cos}(T_i, T_j) = 1 - Dist_c(C_i, C_j) \ , \tag{5.5}$$

where $Dist_c$ is the cosine distance function (see Section 3.2.1). This particular measure is very often used for the comparison of texts due to its robust behaviour in high dimensional and sparse data domains.

The second reference model used in the comparison was also based on the cosine similarity measure. However, unlike in the first one, its similarity judgements were not based on the entire data but were ensembled from 500 local evaluations. Each of those local assessments was made by the cosine similarity restrained to the features selected by a corresponding information bireduct (the same as those used in the construction of the unsupervised RBS model). The similarity function of this model was:

$$Sim_{ens}(T_i, T_j) = \sum_{l=1}^{500} Sim_{cos}(T_i|_{BR_l}, T_j|_{BR_l}), \tag{5.6}$$

were $T|_{BR}$ is a document T represented only by features from BR. This model will be called *cosine ensemble*. It was included in the experiments to investigate the impact of the similarity aggregation technique utilized in unsupervised RBS, on the overall quality of metric-based similarity.

The last reference model, which is called *single RBS*, was constructed using the notion of a commonality relation (Formula 4.12) and the same aggregation method as in the unsupervised RBS (Formula (4.13)). The only difference was that it did not use bireducts to create multiple local sub-models but instead, it made similarity assessments using the whole data set. Such a model can be interpreted as a super-agent whose experience covers all available documents and who takes into consideration all possible factors at once. It was used to verify whether the bireduct-based ensemble approach is beneficial for the unsupervised RBS model.

5.3.3 Results of Experiments

In the experiments, the similarity models described in the previous section were used to assess similarities between every pair of documents from the corpus. This allowed to construct four similarity matrices, in which a value at an intersection of i-th row and j-th column expressed similarity of the document T_i to T_j. The reference semantic proximity matrix was also constructed using Formula (5.3), just as it is described in Section 5.3.1.

Correlations measurements between the values from the similarity matrices obtained for each similarity model and the semantic proximity values are displayed in Table 5.12. Since similarity assessments made using different measures are likely to come from different distributions, the Spearman's rank correlation [148] was utilized in this test to increase its reliability.

Table 5.12: The correlations between the tested similarity models and the semantic proximity.

cosine	cosine ensemble	single RBS	unsupervised RBS
0.155	0.153	0.144	0.186

The result of the unsupervised RBS in this test is much higher than results of other models. It is interesting that the correlation of the third of the reference models (the single RBS) with the semantic proximity was the lowest. This highlights the

benefit from considering multiple similarity aspects in the RBS approach. On the other hand, the difference between the two cosine-base similarity models is negligible which suggests that the ensemble approach may be ineffective for spherical similarity measures.

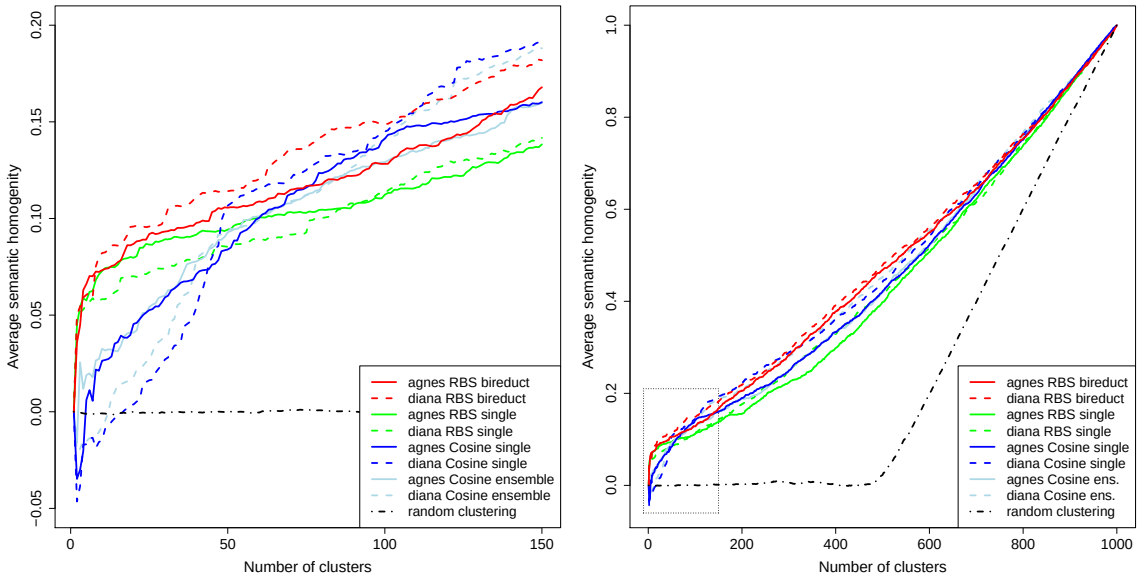


Figure 5.8: The comparison of the average semantic homogeneity of clusterings into a consecutive number of groups using different similarity models and clustering algorithms. The plot on the left is a close up of the most interesting area from the plot on the right. The clustering based on a randomly generated dissimilarity matrix is given as the black dot-dashed line.

The second test involved the computation of two clustering trees for each of the models using the agnes and diana algorithms [78]. Since their implementations from the *cluster* library [121]) can work only with symmetric dissimilarity matrices, the similarity matrix of each examined model had to be transformed using Formula (5.7):

$$dissMatrix = \mathbf{1} - (simMatrix + simMatrix^T)/2 \ , \tag{5.7}$$

where $simMatrix$ is a square matrix with similarity values, $\mathbf{1}$ is a square matrix containing 1 at every position and $*^T$ is the transposition operation.

Figure 5.8 presents average semantic homogeneities (5.5) of clusterings into a consecutive number of groups made using the compared similarity models. The plot on the left is a close up to the area in the plot on the right which is marked by a rectangle with dotted edges. This area corresponds to the most interesting part of the plot because a clustering of documents into a large number of groups produces small individual clusters and is very often difficult to interpret.

The results of this test show evident superiority of the unsupervised RBS similarity over other models for grouping into 2 to 50 groups. Interestingly, in this interval, the semantic homogeneity of the single RBS is also much higher than in the case of the cosine-based measures. The maximum difference between the unsupervised RBS and

the cosine similarity for the agnes algorithm is visible when the clustering is made into 4 groups and is equal to 0.083. For the diana algorithm the difference is even higher – when clustering is made into 10 groups it reaches 0.097. For clustering into 51 to approximately 150 groups the results, especially for the agnes algorithm, change slightly in favour of the cosine similarity. The highest loss of the unsupervised RBS was to the cosine ensemble model and reached 0.015 for division of data into 101 groups. Going further, the unsupervised RBS takes the lead again but the difference is not that apparent. In Figure 5.8 there are also results of a clustering made using a random dissimilarity matrix (as the black dot-dashed line). They can serve as a reference since they clearly show that all of the investigated similarity models led to much better results than the random clusterings.

The compared models differ also in the results of the internal clustering measures. Table 5.13 shows the agglomerative (AC) and divisive (DC) coefficients of the clustering trees obtained for each similarity model.

Table 5.13: Values of the internal clustering separability measures.

measure:	cosine	cosine ensemble	single RBS	unsupervised RBS
AC	0.33	0.37	0.55	0.58
DC	0.28	0.31	0.51	0.54

In this test, the clustering for the both RBS-based models significantly outperformed the cosine similarity approaches. Higher values of the coefficients indicate that the clusterings resulting from the use of the proposed model are clearer (the groups of documents are better separated), thus, they are more likely to be easier to interpret for experts and end-users. It is also worth noticing that the both ensemble-based measures achieved higher internal scores than their single-model counterparts.

Finally, some additional tests were performed to check how some additional information about generated bireducts can be used for selecting relevant local similarity models during the construction of unsupervised RBS. This can be seen as a way of learning an optimal interaction scheme between artificial agents that try to assess the similarity of given documents. In those experiments, the local RBS models were sorted by the decreasing size of the corresponding bireducts[13]. Next, the correlations between the semantic proximity matrix and the similarity assessments (made using an unsupervised RBS model constructed from the first k bireducts) were computed with k ranging from 1 to 500. The highest score was obtained for a model consisting only of 10 bireducts – it reached 0.203 comparing to 0.186 when all the bireducts were used (see Table 5.12). It seems that by using an additional validation document set it would be possible to estimate the optimal number of bireducts to be included into the model, and to increase its overall performance. Moreover, considering a lower number of local models would have a positive impact on the scalability of the proposed similarity learning process.

[13]A size of a bireduct is understood as a sum of cardinalities of its attribute set and its document set.

Chapter 6

Concluding Remarks

This chapter concludes the dissertation and summarises the presented research on similarity learning from high dimensional data. It also indicates some possible research directions for future development of the described models and points out some interesting application areas.

6.1 Summary

The dissertation discusses the problem of learning a similarity relation that reflects human perception and is based on information about exemplary objects represented in an information system. A special focus is on a situation when the considered objects are described by many attributes and thus their typical representation in a metric space would be extremely high dimensional. For such a case, the typically used distance-based similarity models often fail to capture true resemblance between compared objects [15, 159].

Following the research of Amos Tversky on general properties of a similarity relation, a similarity model is proposed in which the metric representation of objects is shifted to a representation by sets of features. In this model, which is called Rule-Based Similarity (RBS), assessments of a similarity degree of a pair of objects depend not only on a context in which the similarity is considered, but also on other objects in the available data. This property remains consistent with observations made by numerous psychologists [41, 42, 51, 83, 159].

The proposed similarity model utilizes notions from the theory of rough sets, which is briefly discussed in Chapter 2. In fact, similarity learning in RBS can be seen as a process of adjusting a similarity approximation space [120, 132, 133] to better fit the desired context. Apart from the fundamental concepts of rough sets, Chapter 2 outlines the rough set approach to selecting relevant attributes (i.e. attribute reduction) and forming rules that represent knowledge about a given data set. Those techniques are later applied in the RBS model for discovering sets of higher-level features that influence similarity judgements.

The third chapter of this dissertation is devoted to the concept of similarity and its general properties. A special emphasis is put on the necessity of fixing a context in which the similarity of two objects is to be considered as it may greatly influence the evaluation outcome (Section 3.1.2). An attempt is also made to form a definition of a

similarity function that would meet intuitive expectations for a natural resemblance measure. As a result, the definition of *a proper similarity function* is proposed in Section 3.1.3, which is followed by a discussion of methods for the evaluation of a similarity function quality. Additionally, Chapter 3 includes an overview of similarity models that are typically employed for solving real-life problems and highlights the essential differences between common distance-based similarity metrics and Tversky's feature contrast model [159]. It also shows application examples of similarity models in a variety of machine learning tasks.

Chapter 4 focuses on techniques that allow learning a similarity relation or a similarity function from data. It starts with an overview of desirable properties of a similarity learning model and a presentation of several approaches to the problem of adjusting a given distance-based similarity function to better fit a data set at hand, by exploiting *the local-global principle* (Section 4.2). Then, Section 4.3 presents the RBS model, which is the main contribution of this dissertation.

The motivation for RBS comes from observations of psychologists who noticed that properties of similarity do not necessarily correspond to those of distance-based models. In fact, in a specific circumstances every basic property of a distance-based similarity relation can be questioned [41, 159]. On the other hand, some practitioners noticed that non-metric representations of objects require defining their higher-level characteristics [7, 51, 103] which often are not present in the original data. For this reason, the construction procedure of RBS, described in Section 4.3.2, includes an automatic feature extraction step that uses decision and inhibitory rules to form sets of arguments for and against the similarity of given objects. During assessment of the similarity, those arguments are aggregated analogically to the contrast model. Unlike in that model, however, weights of the feature sets need not to be set manually, but are derived from available data. Section 4.3.3 discusses several plausible properties of the proposed model and shows that under certain conditions its similarity function meets the definition of the proper similarity function for a similarity in the context of a classification problem.

The original RBS model was extended in order to facilitate its application to two different types of problems that typically involve dealing with high dimensional data. The first extension, which is described in Section 4.3.4, is designed for learning a similarity function in a context of a classification problem from data containing tens thousands of attributes and possibly only a few hundreds of objects. Dynamic Rule-Based Similarity (DRBS) utilizes the notion of dynamic decision reducts for constructing multiple sets of features that may robustly represent different views or aspects of the similarity. Those aspects are then aggregated using DRBS similarity function by an analogy with the Random Forest algorithm [20].

The main purpose of the second extension, called unsupervised RBS (Section 4.3.5), is unsupervised rule-based learning of a semantic resemblance between texts. In order to make it possible, the higher-level features of textual documents that represent relevant aspects of their semantics are extracted using a combination of Explicit Semantic Analysis (ESA) [38, 72] and a novel notion of information bireduct [70, 141, 150]. Due to the utilization of the information bireducts, the evaluation of similarity in the unsupervised RBS model can be interpreted as an interaction between artificial agents who are characterised by different experience and

preferences, and thus have different views on semantics of the compared documents.

Finally, Chapter 5 presents the results of experimental evaluation of different RBS models for a wide array of data types. The performance of the original RBS was compared to several distance-based similarity learning models on well-known benchmark data tables acquired from UCI machine learning repository [36]. The empirical quality evaluation of 1-nearest-neighbour classification revealed that RBS can successfully compete with popular similarity models on standard data sets. For high dimensional microarray data from ArrayExpress [106], not only did DRBS significantly outperform other similarity models but it also achieved better classification results in terms of the balanced accuracy measure than the Random Forest and SVM algorithms, which are considered the state-of-the-art. Unsupervised RBS was also tested and its usefulness for practical applications, such as document clustering, was verified. Groupings constructed using this model turned out to be more semantically homogeneous than those obtained from clustering using standard methods.

6.2 Future Works

There are several possible directions for the future research on the rule-based models of similarity. One idea is to focus on the incorporation of domain knowledge into the model. For example, by using a dedicated similarity ontology it would be possible to model similarity of complex objects or even behavioural patterns changing over time [7, 8, 10, 60]. This kind of a domain knowledge may be effectively used to learn the local similarity relations as well as to create even better higher-level features, e.g. by merging those rules which are semantically similar. Moreover, the method for aggregating arguments for and against the similarity of given objects that is used in RBS is just one of many possibilities. In the future some other aggregation functions could be tried. Such functions could even be learnt from data based on some auxiliary knowledge or interactions with experts.

RBS may also serve as a means for extending notions of rough sets and rough approximations. Currently, there exist several generalizations of rough sets that are based on the notion of similarity [145, 146]. It might be interesting to combine similarity-based rough sets with rough set-based similarity due to the conforming philosophy of those two models. Such a combination can help in obtaining approximations which are more intuitive for human experts and thus can be more useful for real-life data analysis.

Another possible direction in research on RBS is to focus on scalability of the model. In order to facilitate its practical applications in a wide array of domains, scalability of RBS needs to be further enhanced. The scalability can be considered in several aspects, e.g. in terms of a number of training and test objects or in terms of a total number of attributes. Currently, the computational cost of RBS models strongly depends on particular algorithms used for the discretization, attribute reduction and extraction of rules. Having constructed an RBS model, the evaluation of similarity between a single pair of objects can be done in a linear time with regard to the number of extracted rules and objects. Moreover, since a value of RBS similarity function can be computed by a single SQL query, even a sub-linear time complexity

would be possible to achieve by utilization of modern analytical database technologies [124, 143]. Therefore, an implementation of RBS that would be able to make use of contemporary Relational Database Management Systems (RDBMS) would definitely be helpful in real-life applications of the model.

An important factor in the scalability context is also the method for computation of reducts that represent different aspects of the approximated similarity relation. This problem is closely related to an efficient construction of reduct ensembles [141, 144]. The results of the recent research in this topic suggest that an incorporation of auxiliary knowledge about clusterings of original attributes in data can greatly speed up the computation of diverse sets of reducts [69].

Finally, it would be very useful to come up with a unified framework for developing and testing similarity learning methods. Although there exist systems for data analysis that make use of rough set methods for a feature subset selection and extraction of rules, e.g. RSES and RSESlib [12] or Rosetta [56], there is no environment allowing to conveniently combine those tools for the construction of higher-level similarity models. Such an extension, for example in a form of a library for increasingly popular R System [121], would definitely bring benefit to the rough set community, as well as to other data mining researchers. Algorithms used in the construction of RBS models combined with discretization and rule induction methods implemented for the described experiments may serve as a starting point for this task.

Any further progress in the field of learning similarity relation from data would be beneficial to researchers from many domains. This problem is especially important in domains such as biomedicine, where efficient and more accurate models could lead to discovering more effective and safer drugs or better planing of treatments [3, 7, 37, 168]. The classical distance-based approach is often unable to deal with the few-objects-many-attributes problem and the rule-based approach appears to be a promising alternative.

Bibliography

[1] Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *Artificial Intelligence Communications*, 7(1):39–59, 1994.

[2] A An and N Cercone. Rule quality measures for rule induction systems: Description and evaluation. *Computational Intelligence*, 17(3):409–424, 2001.

[3] Pierre Baldi and G. Wesley Hatfield. *DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling*. Cambridge University Press, 2002.

[4] Sugato Basu. *Semi-supervised Clustering: Probabilistic Models, Algorithms and Experiments*. PhD thesis, The University of Texas at Austin, 2005.

[5] Eric Bauer and Ron Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine Learning*, 36(1-2):105–139, 1999.

[6] Jan Bazan. Behavioral pattern identification through rough set modeling. *Fundamenta Informaticae*, 72(1–3):37–50, 2006.

[7] Jan Bazan, Piotr Kruczek, Stanisława Bazan-Socha, Andrzej Skowron, and Jacek J. Pietrzyk. Automatic planning of treatment of infants with respiratory failure through rough set modeling. In *Proceedings of RSCTC 2006*, volume 4259 of *Lecture Notes in Artificial Intelligence*, pages 418–427, Berlin, 2006. Springer. see also the extended version in Fundamenta Informaticae 85, 2008.

[8] Jan Bazan, Sinh Hoa Nguyen, Hung Son Nguyen, and Andrzej Skowron. Rough set methods in approximation of hierarchical concepts. In *Proceedings of RSCTC 2004*, volume 3066 of *Lecture Notes in Artificial Intelligence*, pages 346–355, Berlin, 2004. Springer.

[9] Jan G. Bazan. A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In L. Polkowski and A. Skowron, editors, *Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems*, pages 321–365. Physica Verlag, 1998.

[10] Jan G. Bazan. Transactions on rough sets IX. chapter Hierarchical Classifiers for Complex Spatio-temporal Concepts, pages 474–750. Springer-Verlag, Berlin, Heidelberg, 2008.

[11] Jan G. Bazan, Andrzej Skowron, and Piotr Synak. Dynamic reducts as a tool for extracting laws from decisions tables. In *ISMIS '94: Proceedings of the 8th International Symposium on Methodologies for Intelligent Systems*, pages 346–355, London, UK, 1994. Springer-Verlag.

[12] Jan G. Bazan and Marcin S. Szczuka. RSES and RSESlib - a collection of tools for rough set computations. In Wojciech Ziarko and Y. Y. Yao, editors, *Proceedings of RSCTC 2000*, volume 2005 of *Lecture Notes in Computer Science*, pages 106–113. Springer, 2000.

[13] Richard Beals, David H. Krantz, and Amos Tversky. Foundations of multidimensional scaling. *Psychological Review*, 75(2):127–142, 1968.

[14] Amir Ben-Dor, Laurakay Bruhn, Nir Friedman, Iftach Nachman, Michel Schummer, and Zohar Yakhini. Tissue classification with gene expression profiles. *Journal of Computational Biology*, 7(3-4):559–583, 2000.

[15] Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? In *In International Conference on Database Theory*, pages 217–235, 1999.

[16] Christian Böhm, Christos Faloutsos, and Claudia Plant. Outlier-robust clustering using independent components. In *SIGMOD Conference*, pages 185–198, 2008.

[17] I. Borg and P.J.F. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2005.

[18] Remco R. Bouckaert. Choosing between two learning algorithms based on calibrated tests. In Tom Fawcett and Nina Mishra, editors, *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*, pages 51–58. AAAI Press, 2003.

[19] A. Brazma, P. Hingamp, J. Quackenbush, G. Sherlock, P. Spellman, C. Stoeckert, J. Aach, W. Ansorge, C. A. Ball, H. C. Causton, T. Gaasterland, P. Glenisson, F. C. Holstege, I. F. Kim, V. Markowitz, J. C. Matese, H. Parkinson, A. Robinson, U. Sarkans, S. Schulze-Kremer, J. Stewart, R. Taylor, J. Vilo, and M. Vingron. Minimum Information About a Microarray Experiment (MIAME) - Toward Standards for Microarray Data. *Nature Genetics*, 29(4):365–371, 2001.

[20] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[21] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 539–546, Washington, DC, USA, 2005. IEEE Computer Society.

[22] Vincent Claveau. IRISA Participation in JRS 2012 Data-Mining Challenge: Lazy-Learning with Vectorization. In J.T. Yao et al., editor, *Proceedings of the 8th International Conference on Rough Sets and Current Trends in Computing (RSCTC 2012), Chengdu, China, August 17-20, 2012*, volume 7413 of *Lecture Notes in Artificial Intelligence*, pages 442–449. Springer, Heidelberg, 2012.

[23] D. Coomans and D.L. Massart. Alternative k-nearest neighbour rules in supervised pattern recognition : Part 1. k-nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, 136(0):15–27, 1982.

[24] Peter Dean and A. Famili. Comparative performance of rule quality measures in an induction system. *Applied Intelligence*, 7:113–124, April 1997.

[25] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[26] Pawel Delimata, Mikhail J. Moshkov, Andrzej Skowron, and Zbigniew Suraj. *Inhibitory Rules in Data Analysis: A Rough Set Approach*, volume 163 of *Studies in Computational Intelligence*. Springer, 2009.

[27] Janez Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[28] J. M. Deutsch. Evolutionary algorithms for finding optimal gene sets in microarray prediction. *BMC Bioinformatics*, 19(1):45–52, 2003.

[29] Ramon Diaz-Uriarte and Sara Alvarez de Andres. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(3):online, 2006.

[30] Thomas G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40(2):139–157, 2000.

[31] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. In *Proceedings of the 2003 IEEE Bioinformatics Conference*, pages 523–528, 2003.

[32] Michal Draminski, Marcin Kierczak, Jacek Koronacki, and Henryk Jan Komorowski. Monte Carlo Feature Selection and Interdependency Discovery in Supervised Classification. In *Advances in Machine Learning II*, pages 371–385. 2010.

[33] Didier Dubois and Henri Prade. Rough fuzzy sets and fuzzy rough sets. *International Journal of General Systems*, 17(2-3):191–209, 1990.

[34] Sašo Džeroski, Bojan Cestnik, and Igor Petrovski. Using the m-estimate in rule induction. *Journal of Computing and Information Technology*, 1(1):37–46, March 1993.

[35] Ronen Feldman and James Sanger, editors. *The Text Mining Handbook*. Cambridge University Press, 2007.

[36] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[37] Terrence S. Furey, Nigel Duffy, W. David, and David Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data, 2000.

[38] Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of The Twentieth International Joint Conference for Artificial Intelligence*, pages 1606–1611, Hyderabad, India, 2007.

[39] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, 1998.

[40] Bernhard Ganter, Gerd Stumme, and Rudolf Wille, editors. *Formal Concept Analysis, Foundations and Applications*, volume 3626 of *Lecture Notes in Computer Science*. Springer, 2005.

[41] Itamar Gati and Amos Tversky. Studies of similarity. In E. Rosch and B.B. Lloyd, editors, *Cognition and Categorization*, pages 81–99. L. Erlbaum Associates, Hillsdale, N.J., 1978.

[42] Robert Goldstone, Douglas Medin, and Dedre Gentner. Relational similarity and the nonindependence of features in similarity judgments. *Cognitive Psychology*, 23:222–262, 1991.

[43] Anna Gomolinska. Approximation spaces based on relations of similarity and dissimilarity of objects. *Fundamenta Informaticae*, 79(3-4):319–333, 2007.

[44] Daniel Graupe. *Principles of Artificial Neural Networks*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2nd edition, 2007.

[45] Salvatore Greco, Benedetto Matarazzo, and Roman Słowiński. Fuzzy similarity relation as a basis for rough approximations. In Lech Polkowski and Andrzej Skowron, editors, *Rough Sets and Current Trends in Computing*, volume 1424 of *Lecture Notes in Computer Science*, pages 283–289. Springer, 1998.

[46] Salvatore Greco, Benedetto Matarazzo, and Roman Słowiński. Handling missing values in rough set analysis of multi-attribute and multi-criteria decision problems. In Ning Zhong, Andrzej Skowron, and Setsuo Ohsuga, editors, *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing*, volume 1711 of *Lecture Notes in Computer Science*, pages 146–157. Springer Berlin / Heidelberg, 1999.

[47] Salvatore Greco, Benedetto Matarazzo, and Roman Słowiński. Dominance-based rough set approach to case-based reasoning. In *Modeling Decisions for Artificial Intelligence, Third International Conference, MDAI 2006, Tarragona, Spain, April 3-5, 2006, Proceedings*, volume 3885 of *Lecture Notes in Computer Science*, pages 7–18. Springer, 2006.

[48] Jerzy W. Grzymala-Busse. Rough set strategies to data with missing attribute values. In Tsau Young Lin, Setsuo Ohsuga, Churn-Jung Liau, and Xiaohua Hu, editors, *Foundations and Novel Approaches in Data Mining*, volume 9 of *Studies in Computational Intelligence*, pages 197–212. Springer, 2006.

[49] Jerzy W. Grzymala-Busse and Wojciech Rzasa. Local and global approximations for incomplete data. In Salvatore Greco, Yutaka Hata, Shoji Hirano, Masahiro Inuiguchi, Sadaaki Miyamoto, Hung Son Nguyen, and Roman Słowiński, editors, *Rough Sets and Current Trends in Computing, 5th International Conference, RSCTC 2006, Kobe, Japan, November 6-8, 2006*, pages 244–253, 2006.

[50] Isabelle Guyon and Andre Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[51] Ulrike Hahn and Nick Chater. Understanding similarity: A joint project for psychology, case based reasoning, and law. *Artificial Intelligence Review*, 12:393–427, 1998.

[52] Mark Hall. *Correlation-based Feature Selection for Machine Learning.* PhD thesis, University of Waikato, 1999.

[53] Klaus Hechenbichler and Klaus Schliep. Weighted k-Nearest-Neighbor Techniques and Ordinal Classification, October 2004. a Discussion paper.

[54] Angelos Hliaoutakis, Giannis Varelas, Epimeneidis Voutsakis, Euripides G. M. Petrakis, and Evangelos Milios. Information retrieval by semantic similarity. *Int. Journal on Semantic Web and Information Systems (IJSWIS). Special Issue of Multimedia Semantics*, 3(3):55–73, 2006.

[55] Tu Bao Ho and Ngoc Binh Nguyen. Nonhierarchical document clustering based on a tolerance rough set model. *International Journal of Intelligent Systems*, 17:199–212, 2002.

[56] Aleksander Øhrn and Jan Komorowski. ROSETTA – a rough set toolkit for analysis of data. In *Proceedings Third International Joint Conference on Information Sciences*, pages 403–407, 1997.

[57] Xiaohua Hu and Nick Cercone. Rough sets similarity-based learning from databases. In *KDD*, pages 162–167, 1995.

[58] Edmund Husserl. *The Crisis of European Sciences and Transcendental Phenomenology.* Northwestern University Press, Evanston, 1970. German original written in 1937.

[59] Andrzej Janusz. A similarity relation in machine learning. Master's thesis, University Warsaw, Faculty of Mathematics, Informatics and Mechanics, 2007. In Polish.

[60] Andrzej Janusz. Similarity relation in classification problems. In Chien-Chung Chan, Jerzy W. Grzymala-Busse, and Wojciech P. Ziarko, editors, *Proceedings of RSCTC 2008*, volume 5306 of *Lecture Notes in Artificial Intelligence*, pages 211–222, Heidelberg, 2008. Springer.

[61] Andrzej Janusz. Learning a Rule-Based Similarity: A comparison with the Genetic Approach. In *Proceedings of the Workshop on Concurrency, Specification and Programming (CS&P 2009), Kraków-Przegorzały, Poland, 28-30 September 2009*, volume 1, pages 241–252, 2009.

[62] Andrzej Janusz. Rule-based similarity for classification. In *Proceedings of the WI/IAT 2009 Workshops, 15-18 September 2009, Milan, Italy*, pages 449–452, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[63] Andrzej Janusz. Combining multiple classification or regression models using genetic algorithms. In Marcin S. Szczuka et al., editor, *Proceedings of RSCTC 2010*, volume 6086 of *Lecture Notes in Artificial Intelligence*, pages 130–137, Heidelberg, 2010. Springer.

[64] Andrzej Janusz. Discovering rules-based similarity in microarray data. In *Proceedings of International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2010)*, Lecture Notes in Artificial Intelligence, pages 49–58, Berlin, Heidelberg, 2010. Springer-Verlag.

[65] Andrzej Janusz. Utilization of dynamic reducts to improve performance of the rule-based similarity model for highly-dimensional data. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and International Conference on Intelligent Agent Technology - Workshops*, pages 432–435. IEEE, 2010.

[66] Andrzej Janusz. Combining multiple predictive models using genetic algorithms. *Intelligent Data Analysis*, 16(5):763–776, 2012.

[67] Andrzej Janusz. Dynamic Rule-Based Similarity model for DNA microarray data. *Lecture Notes in Computer Science Transactions on Rough Sets XV*, 7255:1–25, 2012.

[68] Andrzej Janusz, Hung Son Nguyen, Dominik Ślęzak, Sebastian Stawicki, and Adam Krasuski. JRS'2012 Data Mining Competition: Topical Classification of Biomedical Research Papers. In J.T. Yao et al., editor, *Proceedings of the 8th International Conference on Rough Sets and Current Trends in Computing (RSCTC 2012), Chengdu, China, August 17-20, 2012*, volume 7413 of *Lecture Notes in Artificial Intelligence*, pages 417–426. Springer, Heidelberg, 2012.

[69] Andrzej Janusz and Dominik Ślęzak. Utilization of attribute clustering methods for scalable computation of reducts from high-dimensional data. In Maria Ganzha, Leszek A. Maciaszek, and Marcin Paprzycki, editors, *Federated Conference on Computer Science and Information Systems - FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, pages 295–302, 2012.

[70] Andrzej Janusz, Dominik Ślęzak, and Hung Son Nguyen. Unsupervised similarity learning from textual data. *Fundamenta Informaticae*, 119(3).

[71] Andrzej Janusz and Sebastian Stawicki. Applications of approximate reducts to the feature selection problem. In *Proceedings of International Conference on Rough Sets and Knowledge Technology (RSKT)*, volume 6954 of *Lecture Notes in Artificial Intelligence*, pages 45–50. Springer Berlin/Heidelberg, 2011.

[72] Andrzej Janusz, Wojciech Świeboda, Adam Krasuski, and Hung Son Nguyen. Interactive document indexing method based on explicit semantic analysis. In J.T. Yao et al., editor, *Proceedings of the 8th International Conference on Rough Sets and Current Trends in Computing (RSCTC 2012), Chengdu, China, August 17-20, 2012*, volume 7413 of *Lecture Notes in Artificial Intelligence*, pages 156–165. Springer, Heidelberg, 2012.

[73] Richard Jensen and Qiang Shen. New approaches to fuzzy-rough feature selection. *IEEE Transactions on Fuzzy Systems*, 17(4):824–838, 2009.

[74] Thanyaluk Jirapech-Umpai and Stuart Aitken. Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics*, 6(148):online, 2005.

[75] George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant Features and the Subset Selection Problem. In *Proceeding of 11th International Conference on Machine Learning*, pages 121–129. Morgan Kaufmann, 1994.

[76] I. T. Jolliffe. *Principal Component Analysis*. Springer, second edition, October 2002.

[77] Jack David Katzberg and Wojciech Ziarko. Variable precision rough sets with asymmetric bounds. In *Proceedings of the International Workshop on Rough Sets and Knowledge Discovery: Rough Sets, Fuzzy Sets and Knowledge Discovery*, RSKD'93, pages 167–177, London, UK, 1994. Springer-Verlag.

[78] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley Interscience, New York, 1990.

[79] Kenji Kira and Larry A. Rendell. A practical approach to feature selection. In *Proceedings of the ninth international workshop on Machine learning*, ML92, pages 249–256, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[80] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, pages 1137–1145, 1995.

[81] Ron Kohavi and George H. John. Wrappers for feature subset selection. *Artif. Intell.*, 97:273–324, December 1997.

[82] Jan Komorowski, Zdzisław Pawlak, Lech Polkowski, and Andrzej Skowron. Rough sets: A tutorial, 1998.

[83] David H. Krantz and Amos Tversky. Similarity of rectangles: An analysis of subjective dimensions. *Journal of Mathematical Psychology*, 12(1):4–34, 1975.

[84] Nitin Kumar, Nishanth Lolla, Eamonn Keogh, Stefano Lonardi, and Chotirat Ann Ratanamahatana. Time-series bitmaps: a practical visualization tool for working with large time series databases. In *SIAM 2005 Data Mining Conference*, pages 531–535. SIAM, 2005.

[85] Karol Kurach, Krzysztof Pawłowski, Łukasz Romaszko, Marcin Tatjewski, Andrzej Janusz, and Hung Son Nguyen. An ensemble approach to multi-label classification of textual data. In Shuigeng Zhou, Songmao Zhang, and George Karypis, editors, *Advanced Data Mining and Applications*, volume 7713 of *Lecture Notes in Computer Science*, pages 306–317. Springer Berlin Heidelberg, 2012.

[86] Rafal Latkowski. Flexible indiscernibility relations for missing attribute values. *Fundamenta Informaticae*, 67(1-3):131–147, 2005.

[87] Chen Liao, Shutao Li, and Zhiyuan Luo. Gene selection using wilcoxon rank sum test and support vector machine for cancer classification. In Yuping Wang, Yiu-ming Cheung, and Hailin Liu, editors, *Computational Intelligence and Security*, volume 4456 of *Lecture Notes in Computer Science*, pages 57–66. Springer Berlin / Heidelberg, 2007.

[88] T. Marill and D. Green. On the effectiveness of receptors in recognition systems. *IEEE Transactions on Information Theory*, 9(1):11–17, 1963.

[89] Manuel Martín-Merino and Javier Las Rivas. Improving k-nn for human cancer classification using the gene expression profiles. In *IDA '09: Proceedings of the 8th International Symposium on Intelligent Data Analysis*, pages 107–118, Berlin, Heidelberg, 2009. Springer-Verlag.

[90] Andreas Maurer. Learning similarity with operator-valued large-margin classifiers. *Journal of Machine Learning Research*, 9:1049–1082, June 2008.

[91] Isabelle Guyon et al. *Feature Extraction: Foundations and Applications*. Studies in Fuzziness and Soft Computing. Springer, August 2006.

[92] Zbigniew Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1996.

[93] Tom M. Mitchell. *Machine Learning*. McGraw Hill series in computer science. McGraw-Hill, 1997.

[94] Maciej Modrzejewski. Feature selection using rough sets theory. In *Proceedings of the European Conference on Machine Learning*, pages 213–226, London, UK, 1993. Springer-Verlag.

[95] Chi Lang Ngo and Hung Son Nguyen. A tolerance rough set approach to clustering web search results. In Jean-François Boulicaut, Floriana Esposito, Fosca Giannotti, and Dino Pedreschi, editors, *Knowledge Discovery in Databases: PKDD 2004*, volume 3202 of *Lecture Notes in Computer Science*, pages 515–517. Springer Berlin / Heidelberg, 2004.

[96] Hung S. Nguyen. Approximate boolean reasoning: Foundations and applications in data mining. *Transactions on Rough Sets V*, 4100:334–506, 2006.

[97] Hung Son Nguyen. On efficient handling of continuous attributes in large data bases. *Fundamenta Informaticae*, 48(1):61–81, 2001.

[98] Hung Son Nguyen. On the decision table with maximal number of reducts. *Electronic Notes in Theoretical Computer Science*, 82(4):198–205, 2003.

[99] Hung Son Nguyen, Sinh Hoa Nguyen, and Andrzej Skowron. Searching for features defined by hyperplanes. In Zbigniew W. Ras and Maciej Michalewicz, editors, *Foundations of Intelligent Systems, 9th International Symposium, ISMIS '96, Zakopane, Poland, June 9-13, 1996, Proceedings*, volume 1079 of *Lecture Notes in Computer Science*, pages 366–375. Springer, 1996.

[100] Hung Son Nguyen and Andrzej Skowron. Boolean reasoning for feature extraction problems. In Zbigniew W. Ras and Andrzej Skowron, editors, *Foundations of Intelligent Systems, 10th International Symposium, ISMIS '97, Charlotte, North Carolina, USA, October 15-18, 1997, Proceedings*, volume 1325 of *Lecture Notes in Computer Science*, pages 117–126. Springer, 1997.

[101] Hung Son Nguyen and Dominik Ślęzak. Approximate reducts and association rules - correspondence and complexity results. In Ning Zhong, Andrzej Skowron, and Setsuo Ohsuga, editors, *RSFDGrC*, volume 1711 of *Lecture Notes in Computer Science*, pages 137–145. Springer, 1999.

[102] Sin Hoa Thi Nguyen. *Regularity analysis and its applications in data mining.* PhD thesis, Warsaw University, Faculty of Mathematics, Informatics and Mechanics, 1999. Part II: Relational Patterns.

[103] Sinh Hoa Nguyen, Jan Bazan, Andrzej Skowron, and Hung Son Nguyen. Layered learning for concept synthesis. *Lecture Notes in Computer Science Transactions on Rough Sets*, 1(3100):187–208, 2004.

[104] Sankar K. Pal. Soft data mining, computational theory of perceptions, and rough-fuzzy approach. *Information Sciences*, 163(1-3):5–12, 2004.

[105] Sankar K. Pal, Saroj K. Meher, and Soumitra Dutta. Class-dependent rough-fuzzy granular space, dispersion index and classification. *Pattern Recognition*, 45(7):2690–2707, 2012.

[106] Helen E. Parkinson and et al. ArrayExpress update - from an archive of functional genomics experiments to the atlas of gene expression. *Nucleic Acids Research*, 37(Database-Issue):868–872, 2009.

[107] E.A. Patrick and F.P. Fischer III. A generalized k-nearest neighbor rule. *Information and Control*, 16(2):128–152, 1970.

[108] Zdzisław Pawlak. Information systems, theoretical foundations. *Information Systems*, 3(6):205–218, 1981.

[109] Zdzislaw Pawlak. Decision logik. *Bulletin of the EATCS*, 44:201–225, 1991.

[110] Zdzisław Pawlak. *Rough sets - Theoretical Aspects of Reasoning about Data.* Kluwer Academic Publishers, 1991.

[111] Zdzisław Pawlak. Rough sets: present state and the future. *Foundations of Computing and Decision Sciences*, 18(3–4):157–166, 1993.

[112] Zdzisław Pawlak. Rough sets, rough relations and rough functions. *Fundamenta Informaticae*, 27(2-3):103–108, 1996.

[113] Zdzisław Pawlak and Andrzej Skowron. Rough sets and boolean reasoning. *Information Sciences*, 177(1):41–73, 2007.

[114] Zdzisław Pawlak and Andrzej Skowron. Rough sets: Some extensions. *Information Sciences*, 177(1):28–40, 2007.

[115] Zdzisław Pawlak and Andrzej Skowron. Rudiments of rough sets. *Information Sciences*, 177(1):3–27, 2007.

[116] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27:1226–1238, August 2005.

[117] Nada Lavrac Peter, Peter Flach, and Blaz Zupan. *Rule Evaluation Measures: A Unifying View*, pages 174–185. Springer-Verlag, 1999.

[118] Georg Peters, Pawan Lingras, Dominik Ślęzak, and Yiyu Yao. *Rough Sets: Selected Methods and Applications in Management and Engineering.* Advanced Information and Knowledge Processing. Springer London, 2012.

[119] Steven Pinker. *How the mind works*. W. W. Norton, 1998.

[120] Lech T. Polkowski, Andrzej Skowron, and Jan M. Zytkow. Rough foundations for rough sets. In Tsau Young Lin, editor, *Rough Sets and Soft Computing. Conference Proceedings*, pages 142–149, San Jose, California, U.S.A., 1994. San Jose State University.

[121] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008.

[122] Antonio M. Rinaldi. An ontology-driven approach for semantic information retrieval on the web. *ACM Transactions on Internet Technology*, 9:10:1–10:24, July 2009.

[123] Richard J. Roberts. PubMed Central: The GenBank of the published literature. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):381–382, January 2001.

[124] Sunita Sarawagi, Shiby Thomas, and Rakesh Agrawal. Integrating association rule mining with relational database systems: Alternatives and implications. *Data Mining and Knowledge Discovery*, 4(2/3):89–125, 2000.

[125] Roger C. Schank. *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, New York, 1982.

[126] Bernhard Schölkopf. The kernel trick for distances. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pages 301–307. MIT Press, 2000.

[127] Alfred Schütz. *The Phenomenology of the Social World*. Northwestern University Press, Evanston, 1967.

[128] M. Sebag and M. Schoenauer. A rule-based similarity measure. In S. Wess, K. D. Althoff, and M. M. Richter, editors, *Topics in case-based reasoning*, pages 119–130. Springer Verlag, 1994.

[129] Wojciech Siedlecki and Jack Sklansky. Handbook of pattern recognition & computer vision. chapter On automatic feature selection, pages 63–87. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1993.

[130] M. Sikora and B. Sikora. Rough natural hazards monitoring. In Georg Peters, Pawan Lingras, Dominik Ślęzak, and Yiyu Yao, editors, *Selected Methods and Applications of Rough Sets in Management and Engineering*, Advanced Information and Knowledge Processing, pages 163–179. Springer London, 2012.

[131] Andrzej Skowron and Cecylia Rauszer. *The Discernibility Matrices and Functions in Information Systems*, pages 331–362. Kluwer, Dordrecht, 1992.

[132] Andrzej Skowron and Jarosław Stepaniuk. Approximation of relations. In *RSKD'93: Proceedings of the International Workshop on Rough Sets and Knowledge Discovery*, pages 161–166, London, UK, 1994. Springer-Verlag.

[133] Andrzej Skowron and Jarosław Stepaniuk. Tolerance approximation spaces. *Fundamenta Informaticae*, 27(2/3):245–253, 1996.

[134] Andrzej Skowron, Jarosław Stepaniuk, James F. Peters, and Roman W. Swiniarski. Calculi of approximation spaces. *Fundamenta Informaticae*, 72(1-3):363–378, 2006.

[135] Andrzej Skowron, Jarosław Stepaniuk, and Roman W. Świniarski. Modeling rough granular computing based on approximation spaces. *Information Sciences*, 184(1):20–43, 2012.

[136] Dominik Ślęzak. Approximate reducts in decision tables. In *Proceedings of IPMU'1996*, 1996.

[137] Dominik Ślęzak. *Various approaches to reasoning with frequency based decision reducts: a survey*, pages 235–285. Physica-Verlag GmbH, Heidelberg, Germany, Germany, 2000.

[138] Dominik Ślęzak. Approximate entropy reducts. *Fundamenta Informaticae*, 53(3-4):365–390, 2002.

[139] Dominik Ślęzak. Rough sets and few-objects-many-attributes problem: The case study of analysis of gene expression data sets. *Frontiers in the Convergence of Bioscience and Information Technologies*, 0:437–442, 2007.

[140] Dominik Ślęzak. Rough sets and functional dependencies in data: Foundations of association reducts. In Marina Gavrilova, C. Tan, Yingxu Wang, and Keith Chan, editors, *Transactions on Computational Science V*, volume 5540 of *Lecture Notes in Computer Science*, pages 182–205. Springer Berlin / Heidelberg, 2009.

[141] Dominik Ślęzak and Andrzej Janusz. Ensembles of bireducts: Towards robust classification and simple representation. In Tai-Hoon Kim, Hojjat Adeli, Dominik Ślęzak, Frode Eika Sandnes, Xiaofeng Song, Kyo-Il Chung, and Kirk P. Arnett, editors, *Future Generation Information Technology - Third International Conference, FGIT 2011 in Conjunction with GDC 2011, Jeju Island, Korea, December 8-10, 2011. Proceedings*, volume 7105 of *Lecture Notes in Computer Science*, pages 64–77. Springer, 2011.

[142] Dominik Ślęzak, Andrzej Janusz, Wojciech Świeboda, Hung Son Nguyen, Jan G. Bazan, and Andrzej Skowron. Semantic analytics of PubMed content. In Andreas Holzinger and Klaus-Martin Simonic, editors, *Information Quality in e-Health - 7th Conference of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2011, Graz, Austria, November 25-26, 2011. Proceedings*, volume 7058 of *Lecture Notes in Computer Science*, pages 63–74. Springer, 2011.

[143] Dominik Ślęzak, Piotr Synak, Janusz Borkowski, Jakub Wroblewski, and Graham Toppin. A rough-columnar rdbms engine – a case study of correlated subqueries. *IEEE Data Engineering Bulletin*, 35(1):34–39, 2012.

[144] Dominik Ślęzak and Sebastian Widz. Is it important which rough-set-based classifier extraction and voting criteria are applied together? In *Proceedings of International Conference on Rough Sets and Current Trends in Computing (RSCTC)*, volume 6086 of *Lecture Notes in Artificial Intelligence*, pages 187–196. Springer, 2010.

[145] Roman Słowiński and D. Vanderpooten. Similarity relation as a basis for rough approximations. In P.P. Wang, editor, *Advances in Machine Intelligence and Soft-Computing, vol.IV*, pages 17–33. Duke University Press, Durham, NC, 1997.

[146] Roman Słowiński and D. Vanderpooten. A generalized definition of rough approximations based on similarity. *IEEE Transactions on Data and Knowledge Engineering*, 12:331–336, 2000.

[147] Barry Smyth and Paul McClave. Similarity vs. diversity. In *Proceedings of ICCBR 2001*, volume 2080 of *Lecture Notes in Artificial Intelligence*, pages 347–361, Berlin, 2001. Springer.

[148] C. Spearman. The proof and measurement of association between two things. By C. Spearman, 1904. *The American Journal of Psychology*, 100(3-4):441–471, 1987.

[149] Armin Stahl and Thomas Gabel. Using evolution programs to learn local similarity measures. In *In Proceedings of the Fifth International Conference on Case-Based Reasoning*, pages 537–551. Springer, 2003.

[150] Sebastian Stawicki and Sebastian Widz. Decision bireducts and approximate decision reducts: Comparison of two approaches to attribute subset ensemble construction. In Maria Ganzha, Leszek A. Maciaszek, and Marcin Paprzycki, editors, *Federated Conference on Computer Science and Information Systems - FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, pages 331–338, 2012.

[151] Jerzy Stefanowski. An experimental study of methods combining multiple classifiers - diversified both by feature selection and bootstrap sampling. In Krassimir T. Atanassov, Janusz Kacprzyk, Maciej Krawczak, and Eulalia Szmidt, editors, *Issues in the Representation and Processing of Uncertain and Imprecise Information*, pages 337–354. Akademicka Oficyna Wydawnicza EXIT, Warsaw, 2005.

[152] Jerzy Stefanowski and Alexis Tsoukiàs. Incomplete information tables and rough classification. *Computational Intelligence*, 17(3):545–566, 2001.

[153] Grant Strong and Minglun Gong. Similarity-based image organization and browsing using multi-resolution self-organizing map. *Image Vision Comput.*, 29(11):774–786, 2011.

[154] Wojciech Świeboda and Hung Son Nguyen. Rough Set Methods for Large and Sparse Data in EAV Format. In *2012 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), Ho Chi Minh City, Vietnam, February 27 - March 1, 2012*, pages 1–6. IEEE, 2012.

[155] Marcin S. Szczuka, Andrzej Janusz, and Kamil Herba. Clustering of rough set related documents with use of knowledge from DBpedia. In Jingtao Yao, Sheela Ramanna, Guoyin Wang, and Zbigniew Suraj, editors, *Rough Sets and Knowledge Technology*, volume 6954 of *Lecture Notes in Computer Science*, pages 394–403. Springer Berlin/Heidelberg, 2011.

[156] Marcin S. Szczuka, Andrzej Skowron, and Jarosław Stepaniuk. Function approximation and quality measures in rough-granular systems. *Fundamenta Informaticae*, 109(3):339–354, 2011.

[157] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison Wesley, Boston, 2006.

[158] Paul Thagard. *Mind: Introduction to Cognitive Science*, chapter 10. MIT Press, Cambridge, Massachusetts, segunda edition, 2005.

[159] Amos Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

[160] Amos Tversky and David H. Krantz. The dimensional representation and the metric structure of similarity data. *Journal of Mathematical Psychology*, 7(3):572–596, 1970.

[161] United States National Library of Medicine. Introduction to MeSH - 2011. http://www.nlm.nih.gov/mesh/introduction.html, 2011.

[162] Julio Valdés and Alan Barton. Relevant attribute discovery in high dimensional data: Application to breast cancer gene expressions. pages 482–489. 2006.

[163] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., NY, USA, 1995.

[164] S.S. Vempala. *The Random Projection Method*. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, 2004.

[165] A. W. Whitney. A Direct Method of Nonparametric Measurement Selection. *IEEE Transactions on Computers*, 20:1100–1103, September 1971.

[166] Arkadiusz Wojna. *Analogy-based reasoning in classifier construction*. PhD thesis, Warsaw University, Faculty of Mathematics, Informatics and Mechanics, 2004.

[167] Marcin Wojnarski. LTF-C: Architecture, training algorithm and applications of new neural classifier. *Fundamenta Informaticae*, 54(1):89–105, 2003.

[168] Marcin Wojnarski, Andrzej Janusz, Hung Son Nguyen, Jan Bazan, ChuanJiang Luo, Ze Chen, Feng Hu, Guoyin Wang, Lihe Guan, Huan Luo, Juan Gao, Yuanxia Shen, Vladimir Nikulin, Tian-Hsiang Huang, Geoffrey J. McLachlan, Matko Bošnjak, and Dragan Gamberger. RSCTC'2010 discovery challenge: Mining DNA microarray data for medical diagnosis and treatment. In Marcin S. Szczuka et al., editor, *Proceedings of RSCTC 2010*, volume 6086 of *Lecture Notes in Artificial Intelligence*, pages 4–19, Heidelberg, 2010. Springer.

[169] Jakub Wróblewski. Ensembles of classifiers based on approximate reducts. *Fundamenta Informaticae*, 47(3-4):351–360, October 2001.

[170] Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart J. Russell. Distance metric learning with application to clustering with side-information. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems 15, NIPS 2002, December 9-14, 2002, Vancouver, British Columbia, Canada*, pages 505–512. MIT Press, 2002.

[171] Huilin Xiong and Xue-wen Chen. Kernel-based distance metric learning for microarray data classification. *BMC Bioinformatics*, 7(299):online, 2006.

[172] Yiyu Yao. Semantics of fuzzy sets in rough set theory. *Transactions on Rough Sets II*, 3135:297–318, 2004.

[173] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[174] Ning Zhong, Juzhen Dong, and Setsuo Ohsuga. Using rough sets with heuristics for feature selection. *Journal of Intelligent Information Systems*, 16(3):199–214, October 2001.

[175] Wojciech Ziarko. Variable precision rough set model. *Journal of Computer and System Sciences*, 46:39–59, February 1993.