Fuzzy Rough Sets: from Theory into Practice

Chris Cornelis[1], Martine De Cock[1], Anna Maria Radzikowska[2]
[1] Computational Web Intelligence
Dept. of Applied Mathematics and Computer Science
Ghent University, Krijgslaan 281 (S9), 9000 Gent, Belgium
{Chris.Cornelis, Martine.DeCock}@UGent.be
http://www.cwi.UGent.be
[2] Faculty of Mathematics and Information Science
Warsaw University of Technology,
Plac Politechniki 1, 00-661 Warsaw, Poland
annrad@mini.pw.edu.pl

Abstract

Fuzzy sets and rough sets address two important, and mutually orthogonal, characteristics of imperfect data and knowledge: while the former allow that objects belong to a set or relation to a given degree, the latter provide approximations of concepts in the presence of incomplete information. In this chapter, we demonstrate how these notions can be combined into a hybrid theory that is able to capture the best of different worlds. In particular, we review various alternatives for defining lower and upper approximations of a fuzzy set under a fuzzy relation, and also explore their application in query refinement.

1 Introduction

Fuzzy sets (Zadeh [35], 1965), as well as the slightly younger rough sets (Pawlak [23], 1982), have left an important mark on the way we represent and compute with imperfect information nowadays. Each of them has fostered a broad research community, and their impact has also been clearly felt at the application level. Although it was recognized early on that the associated theories are complementary rather than competitive, perceived similarities between both concepts and efforts to prove that one of them subsumes the other, have somewhat stalled progress towards shaping a hybrid theory that combines their mutual strengths.

Still, seminal research on fuzzy rough set theory flourished during the 1990's and early 2000's (e.g. [10, 15, 16, 18, 20, 21, 27, 30, 34]), and recently, cross-disciplinary research has also profited from the popularization and widespread adoption of two important computing paradigms: granular computing, with its focus on clustering information entities into granules in terms of similarity, indistinguishability, . . . has helped the theoretical underpinnings of the hybrid theory to come of age, while soft computing—a collection of techniques that are tolerant of typical characteristics of imperfect data and knowledge, and hence adhere closer to the human mind than conventional hard computing techniques—has stressed the role of fuzzy sets and rough sets as partners, rather than adversaries, within a panoply of practical applications.

Within the hybrid theory, Pawlak's well-known framework for the construction of lower and upper approximations of a concept C given incomplete information (a subset A of a given universe X, containing examples of C), and an equivalence relation R in X that models "indiscernibility" or "indistinguishability", has been extended in two ways:

1. The set A may be generalized to a fuzzy set in X, allowing that objects can belong to a concept (i.e., meet its characteristics) to varying degrees.

2. Rather than modeling elements' indistinguishability, we may assess their similarity (objects are similar to a certain degree), represented by a fuzzy relation R. As a result, objects are categorized into classes, or granules, with "soft" boundaries based on their similarity to one another.

In this paper, we consider the general problem of defining lower and upper approximations of a fuzzy set A by means of a fuzzy relation R. A key ingredient to our exposition will be the fact that elements of X can belong, to varying degrees, to several "soft granules" simultaneously. Not only does this property lie right at the heart of fuzzy set theory, a similar phenomenon can already be observed in crisp, or traditional, rough set theory as soon as the assumption that R is an equivalence relation (and hence induces a partition of X) is abandoned. Within fuzzy rough set theory, the impact of this property—which plays a crucial role towards defining the approximations—is felt still more strongly, since even fuzzy $\mathcal{T}$-equivalence relations, the natural candidates for generalizing equivalence relations, are subject to it.

The paper is structured as follows. In Section 2, we first recall the necessary background on rough sets and fuzzy sets. Section 3 reviews various proposals for the definition of a fuzzy rough set and examines their respective properties. Furthermore, Section 4 reveals that the various alternative definitions are not just of theoretical interest but become useful in a topical application such as query refinement for searching on the WWW, especially in the presence of ambiguous query terms.

2 Preliminaries

2.1 Rough Sets

Rough set analysis makes statements about the membership of some element y of X to the concept of which A is a set of examples, based on the indistinguishability between y and the elements of A. Usually, indistinguishability is described by means of an equivalence relation R on X; for example, if the elements of X are represented by a set of attributes, two elements of X are indistinguishable if they have the same value for all attributes. In this case, (X, R) is called a standard, or Pawlak, approximation space. More generally, it is possible to replace R by any binary relation in X, not necessarily an equivalence relation; we then call (X, R) a generalized approximation space. In particular, the case of a reflexive R, and of a tolerance, i.e. reflexive and symmetric, relation R have received ample attention in the literature.

In all cases, A is approximated in two ways, resulting in the lower and upper approximation of the concept. In the next paragraphs, we will review the definitions of these approximations.

For completeness we mention that a second stream concerning rough sets in the literature was initiated by Iwinski [14] who did not use an equivalence relation or tolerance relation as an initial building block to define the rough set concept. Although this formulation provides an elegant mathematical model, the absence of the equivalence relation makes his model hard to interpret. We therefore do not deal with it in this chapter; a more detailed comparison of the different views on rough set theory can be found in e.g. [33].

2.1.1 Rough Sets in Pawlak Approximation Spaces

In a Pawlak approximation space (X, R), an element y of X belongs to the lower approximation $R \downarrow A$ of A if the equivalence class to which y belongs is included in A. On the other hand y belongs to the upper approximation $R \uparrow A$ of A if its equivalence class has a non-empty intersection with A. Formally, the sets $R{\downarrow}A$ and $R{\uparrow}A$ are defined by, for y in X,

$$y \in R{\downarrow}A \quad \text{iff} \quad [y]_R \subseteq A \tag{1}$$

$$y \in R{\uparrow}A \quad \text{iff} \quad [y]_R \cap A \neq \emptyset \tag{2}$$

In other words

$$y \in R{\downarrow}A \quad \text{iff} \quad (\forall x \in X)((x, y) \in R \Rightarrow x \in A) \tag{3}$$

$$y \in R{\uparrow}A \quad \text{iff} \quad (\exists x \in X)((x, y) \in R \wedge x \in A) \tag{4}$$

The underlying meaning is that $R{\downarrow}A$ is the set of elements *necessarily* satisfying the concept (strong membership), while $R{\uparrow}A$ is the set of elements *possibly* belonging to the concept (weak membership).

Some basic and easily verified properties of lower and upper approximation are summarized in Table 1. From 2., it holds that $R{\downarrow}A \subseteq R{\uparrow}A$. If y belongs to the boundary region $R{\uparrow}A \backslash R{\downarrow}A$, then there is some doubt, because in this case y is at the same time indistinguishable from at least one element of A and at least one element of X that is not in A. Following [27], we call (A_1, A_2) a rough set (in (X, R)) as soon as there is a set A in X such that $R{\downarrow}A = A_1$ and $R{\uparrow}A = A_2$.

Table 1: Properties of lower and upper approximation in a Pawlak approximation space (X, R); A and B are subsets of X, and co denotes set-theoretic complement.

1.	$R{\uparrow}A = co(R{\downarrow}(coA))$ $R{\downarrow}A = co(R{\uparrow}(coA))$
2.	$R{\downarrow}A \subseteq A \subseteq R{\uparrow}A$
3.	$A \subseteq B \Rightarrow (R{\downarrow}A \subseteq R{\downarrow}B$ and $R{\uparrow}A \subseteq R{\uparrow}B)$
4.	$R{\downarrow}(A \cap B) = R{\downarrow}A \cap R{\downarrow}B$ $R{\uparrow}(A \cap B) \subseteq R{\uparrow}A \cap R{\uparrow}B$
5.	$R{\downarrow}(A \cup B) \supseteq R{\downarrow}A \cup R{\downarrow}B$ $R{\uparrow}(A \cup B) = R{\uparrow}A \cup R{\uparrow}B$
6.	$R{\downarrow}(R{\downarrow}A) = R{\downarrow}A$ $R{\uparrow}(R{\uparrow}A) = R{\uparrow}A$

2.1.2 Rough Sets in Generalized Approximation Spaces

For an arbitrary binary relation R in X, the role of equivalence classes in Pawlak approximation spaces (cfr. formulas (1) and (2)) can be subsumed by the more general concept of R-foresets; recall that, for y in X, the R-foreset Ry is defined by

$$Ry = \{x \mid x \in X \text{ and } (x, y) \in R\} \tag{5}$$

It is well known that for an equivalence relation R, then R induces a partition of X, so if we consider two equivalence classes then they either coincide or are disjoint. It is therefore not possible for y to belong to two different equivalence classes at the same time. If R is a non-equivalence relation in X, however, then it is quite normal that different foresets may partially overlap.

By the definition used so far, y belongs to the lower approximation of A if Ry is included in A. In view of the discussion above however it makes sense to consider also other R-foresets that contain y, and to assess their inclusion into A as well for the lower approximation, and their overlap with A for the upper approximation. This idea, first explored by Pomykala [25], results in the following (inexhaustive!) list of candidate definitions for the lower and the upper approximation of A:

1. y belongs to the lower approximation of A iff
   (a) all R-foresets containing y are included in A
   (b) at least one R-foreset containing y is included in A
   (c) Ry is included in A
2. y belongs to the upper approximation of A iff

(a) all R-foresets containing y have a non-empty intersection with A

(b) at least one R-foreset containing y has a non-empty intersection with A

(c) Ry has a non-empty intersection with A

Paraphrasing these expressions, we obtain the following definitions:

1. The tight, loose and (usual) lower approximation of A are defined as

   (a) $y \in R{\downarrow}{\downarrow}A$ iff $(\forall z \in X)(y \in Rz \Rightarrow Rz \subseteq A)$

   (b) $y \in R{\uparrow}{\downarrow}A$ iff $(\exists z \in X)(y \in Rz \wedge Rz \subseteq A)$

   (c) $y \in R{\downarrow}A$ iff $Ry \subseteq A$

   for all y in X.

2. The tight, loose and (usual) upper approximation of A are defined as

   (a) $y \in R{\downarrow}{\uparrow}A$ iff $(\forall z \in X)(y \in Rz \Rightarrow Rz \cap A \neq \emptyset)$

   (b) $y \in R{\uparrow}{\uparrow}A$ iff $(\exists z \in X)(y \in Rz \wedge Rz \cap A \neq \emptyset)$

   (c) $y \in R{\uparrow}A$ iff $Ry \cap A \neq \emptyset$

   for all y in X.

Note 1 The terminology "tight" refers to the fact that we take all R-foresets classes into account, giving rise to a strict or tight requirement. For the "loose" approximations, we only look at "the best one" which is clearly a more flexible demand. For an equivalence relation R, all of the above definitions coincide, but in general they can be different as the following example shows.

Example 2 Consider $X = \{x_1, x_2, x_3, x_4\}$, $A = \{x_1, x_3\}$ and the relation R in X defined by

R	x_1	x_2	x_3	x_4
x_1	1	0	1	0
x_2	1	1	0	1
x_3	0	1	1	0
x_4	1	1	0	1

Then

$$\begin{array}{rclrcl} R\downarrow A & = & \{x_3\} & R\uparrow A & = & \{x_1, x_2, x_3\} \\ R\uparrow\downarrow A & = & \{x_1, x_3\} & R\uparrow\uparrow A & = & X \\ R\downarrow\downarrow A & = & \emptyset & R\downarrow\uparrow A & = & \{x_1, x_3\} \end{array}$$

In the remainder of this section, we assume that R is reflexive and symmetric, which are basic requirements if R is supposed to model similarity. The symmetry of R allows to verify following relationships between the approximations:

$$\begin{aligned} R{\downarrow}{\downarrow}A &= R{\downarrow}(R{\downarrow}A) && (6)\\ R{\uparrow}{\downarrow}A &= R{\uparrow}(R{\downarrow}A) && (7)\\ R{\downarrow}{\uparrow}A &= R{\downarrow}(R{\uparrow}A) && (8)\\ R{\uparrow}{\uparrow}A &= R{\uparrow}(R{\uparrow}A) && (9) \end{aligned}$$

Table 2 lists the properties of the different approximations. Interesting observations to make from this table include:

1. By 1., there are three pairs of dual approximation operators w.r.t. complementation.
2. Property 2. shows the relationship between the approximations in terms of inclusion, and how A itself fits into this picture. Note how these relationships nicely justify the terminology.
3. Loose lower, resp. tight upper, approximation satisfies only a weak interaction property w.r.t. set intersection, resp. union (Property 4. and 5.).
4. By Property 6. of Table 1, when R is an equivalence relation, lower and upper approximation are idempotent. This means that in Pawlak approximation spaces, maximal reduction and expansion are achieved within one approximation step. The same holds true for loose lower and tight upper approximation in a symmetric approximation space, but not for the other operators; for these, a gradual reduction/expansion process is obtained by successively taking approximations.

2.2 Fuzzy Sets

In the context of fuzzy rough set theory, A is a fuzzy set in X, i.e. an $X \to [0,1]$ mapping, while R is a fuzzy relation in X, i.e. a fuzzy set in $X \times X$. Recall that for all y in X, the R-foreset of y is the fuzzy set Ry defined by

$$Ry(x) = R(x,y) \tag{10}$$

for all x in X. The fuzzy logical counterparts of the connectives in (3) and (4) play an important role in the generalization of lower and upper approximations; we therefore recall some important definitions.

First, a negator $\mathcal{N}$ is a decreasing $[0,1] \to [0,1]$ mapping satisfying $\mathcal{N}(0) = 1$ and $\mathcal{N}(1) = 0$. $\mathcal{N}$ is called involutive if $\mathcal{N}(\mathcal{N}(x)) = x$ for all x in $[0,1]$. The standard negator

Table 2: Properties of lower and upper approximation in a symmetric approximation space (X, R).

1.	$R{\uparrow}A = co(R{\downarrow}(coA))$	4.	$R{\downarrow}(A \cap B) = R{\downarrow}A \cap R{\downarrow}B$
	$R{\downarrow}A = co(R{\uparrow}(coA))$		$R{\uparrow}(A \cap B) \subseteq R{\uparrow}A \cap R{\uparrow}B$
	$R{\downarrow}{\uparrow}A = co(R{\uparrow}{\downarrow}(coA))$		$R{\downarrow}{\uparrow}(A \cap B) \subseteq R{\downarrow}{\uparrow}A \cap R{\downarrow}{\uparrow}B$
	$R{\uparrow}{\downarrow}A = co(R{\downarrow}{\uparrow}(coA))$		$R{\uparrow}{\downarrow}(A \cap B) \subseteq R{\uparrow}{\downarrow}A \cap R{\uparrow}{\downarrow}B$
	$R{\uparrow}{\uparrow}A = co(R{\downarrow}{\downarrow}(coA))$		$R{\uparrow}{\uparrow}(A \cap B) \subseteq R{\uparrow}{\uparrow}A \cap R{\uparrow}{\uparrow}B$
	$R{\downarrow}{\downarrow}A = co(R{\uparrow}{\uparrow}(coA))$		$R{\downarrow}{\downarrow}(A \cap B) = R{\downarrow}{\downarrow}A \cap R{\downarrow}{\downarrow}B$
2.	$R{\downarrow}{\downarrow}A \subseteq R{\downarrow}A \subseteq R{\uparrow}{\downarrow}A \subseteq A$	5.	$R{\downarrow}(A \cup B) \supseteq R{\downarrow}A \cup R{\downarrow}B$
	$A \subseteq R{\downarrow}{\uparrow}A \subseteq R{\uparrow}A \subseteq R{\uparrow}{\uparrow}A$		$R{\uparrow}(A \cup B) = R{\uparrow}A \cup R{\uparrow}B$
			$R{\downarrow}{\uparrow}(A \cup B) \supseteq R{\downarrow}{\uparrow}A \cup R{\downarrow}{\uparrow}B$
3.	$A \subseteq B \Rightarrow \begin{cases} R{\downarrow}A \subseteq R{\downarrow}B \\ R{\uparrow}A \subseteq R{\uparrow}B \\ R{\downarrow}{\uparrow}A \subseteq R{\downarrow}{\uparrow}B \\ R{\uparrow}{\downarrow}A \subseteq R{\uparrow}{\downarrow}B \\ R{\uparrow}{\uparrow}A \subseteq R{\uparrow}{\uparrow}B \\ R{\downarrow}{\downarrow}A \subseteq R{\downarrow}{\downarrow}B \end{cases}$		$R{\uparrow}{\downarrow}(A \cup B) \supseteq R{\uparrow}{\downarrow}A \cup R{\uparrow}{\downarrow}B$ $R{\uparrow}{\uparrow}(A \cup B) = R{\uparrow}{\uparrow}A \cup R{\uparrow}{\uparrow}B$ $R{\downarrow}{\downarrow}(A \cup B) \supseteq R{\downarrow}{\downarrow}A \cup R{\downarrow}{\downarrow}B$
		6.	$R{\downarrow}{\uparrow}(R{\downarrow}{\uparrow}A) = R{\downarrow}{\uparrow}A$ $R{\uparrow}{\downarrow}(R{\uparrow}{\downarrow}A) = R{\uparrow}{\downarrow}A$

$\mathcal{N}_s$ is defined by $\mathcal{N}_s(x) = 1-x$. A negator $\mathcal{N}$ induces a corresponding fuzzy set complement $co_{\mathcal{N}}$: for any fuzzy set A in X and every element x in X,

$$co_{\mathcal{N}}(A) = \mathcal{N}(A(x)) \tag{11}$$

A triangular norm (t-norm for short) $\mathcal{T}$ is any increasing, commutative and associative $[0,1]^2 \to [0,1]$ mapping satisfying $\mathcal{T}(1,x) = x$, for all x in $[0,1]$. Analogously, a triangular conorm (t-conorm for short) $\mathcal{S}$ is any increasing, commutative and associative $[0,1]^2 \to [0,1]$ mapping satisfying $\mathcal{S}(0,x) = x$, for all x in $[0,1]$. Table 3 mentions some important t-norms and t-conorms. The $\mathcal{T}$-intersection and $\mathcal{S}$-union of fuzzy sets A and B in X are defined by

$$(A \cap_{\mathcal{T}} B)(x) = \mathcal{T}(A(x), B(x)) \tag{12}$$

$$(A \cap_{\mathcal{S}} B)(x) = \mathcal{S}(A(x), B(x)) \tag{13}$$

for all x in X. Throughout this paper, $A \cap_{\mathcal{T}_\mathrm{M}} B$ and $A \cup_{\mathcal{S}_\mathrm{M}} B$ are abbreviated to $A \cap B$ and $A \cup B$ and called standard complement and union, respectively.

Finally, an implicator is any $[0,1]^2 \to [0,1]$-mapping $\mathcal{I}$ satisfying $\mathcal{I}(0,0) = 1, \mathcal{I}(1,x) = x$, for all x in $[0,1]$. Moreover we require $\mathcal{I}$ to be decreasing in its first, and increasing in its second component. If $\mathcal{T}$ is a t-norm, the mapping $\mathcal{I}_{\mathcal{T}}$ defined by, for all x and y in [0,1],

$$\mathcal{I}_{\mathcal{T}}(x,y) = \sup\{\lambda | \lambda \in [0,1] \text{ and } \mathcal{T}(x,\lambda) \leq y\} \tag{14}$$

is an implicator, usually called the residual implicator of $\mathcal{T}$. If $\mathcal{T}$ is a t-norm and $\mathcal{N}$ is an involutive negator, then the mapping $\mathcal{I}_{\mathcal{T},\mathcal{N}}$ defined by, for all x and y in [0,1],

$$\mathcal{I}_{\mathcal{T},\mathcal{N}}(x,y) = \mathcal{N}(\mathcal{T}(x,\mathcal{N}(y))) \tag{15}$$

is an implicator, usually called the S-implicator induced by $\mathcal{T}$ and $\mathcal{N}$. In Table 4, we mention some important S- and residual implicators; the S-implicators are induced by means of the standard negator $\mathcal{N}_s$.

Table 3: Well-known t-norms and t-conorms; x and y in $[0,1]$.

t-norm	t-conorm
$\mathcal{T}_{\mathrm{M}}(x,y) = \min(x,y)$	$\mathcal{S}_{\mathrm{M}}(x,y) = \max(x,y)$
$\mathcal{T}_{\mathrm{P}}(x,y) = xy$	$\mathcal{S}_{\mathrm{P}}(x,y) = x+y-xy$
$\mathcal{T}_{\mathrm{W}}(x,y) = \max(x+y-1,0)$	$\mathcal{S}_{\mathrm{W}}(x,y) = \min(x+y,1)$

Table 4: Well-known implicators; x and y in $[0,1]$.

S-implicator	Residual implicator
$\mathcal{I}_{\mathcal{S}_{\mathrm{M}}}(x,y) = \max(1-x,y)$	$\mathcal{I}_{\mathcal{T}_{\mathrm{M}}}(x,y) = \begin{cases} 1, & \text{if } x \le y \\ y, & \text{otherwise} \end{cases}$
$\mathcal{I}_{\mathcal{S}_{\mathrm{P}}}(x,y) = 1-x+xy$	$\mathcal{I}_{\mathcal{T}_{\mathrm{P}}}(x,y) = \begin{cases} 1, & \text{if } x \le y \\ \frac{y}{x}, & \text{otherwise} \end{cases}$
$\mathcal{I}_{\mathcal{S}_{\mathrm{W}}}(x,y) = \min(1-x+y,1)$	$\mathcal{I}_{\mathcal{S}_{\mathrm{W}}}(x,y) = \min(1-x+y,1)$

In fuzzy rough set theory, we require a way to express that objects are similar to each other to some extent. In the context of this paper, similarity is modelled by a fuzzy tolerance relation R, that is

$$\begin{array}{ll} R(x,x) = 1 & \text{(reflexivity)} \\ R(x,y) = R(y,x) & \text{(symmetry)} \end{array}$$

hold for all x and y in X. Additionally, $\mathcal{T}$-transitivity (for a particular t-norm $\mathcal{T}$) is sometimes imposed: for all x, y and z in X,

$$\mathcal{T}(R(x,y),R(y,z)) \le R(x,z) \quad (\mathcal{T}\text{-transitivity})$$

R is then called a fuzzy $\mathcal{T}$-equivalence relation; because equivalence relations are used to model equality, fuzzy $\mathcal{T}$-equivalence relations are commonly considered to represent approximate equality. In general, for a fuzzy tolerance relation R, we will call Ry the "fuzzy similarity class" of y.

3 Fuzzy Rough Sets

3.1 Definitions

Research on fuzzifying lower and upper approximations in the spirit of Pawlak emerged in the late 1980's. Chronologically, the first proposals are due to Nakamura [20], and to Dubois and Prade [10] who drew inspiration from an earlier publication by Fariñas del Cerro and Prade [11].

In developing the generalizations, the central focus moved from elements' indistinguishability (for instance, w.r.t. their attribute values in an information system) to their similarity: objects are categorized into classes with "soft" boundaries based on their similarity to one another. A concrete advantage of such a scheme is that abrupt transitions between classes are replaced by gradual ones, allowing that an element can belong (to varying degrees) to more than one class. An example at hand is an attribute "age" in an information table: in order to restrict the number of equivalence classes, classical rough set theory advises to discretize age values by a crisp partition of the universe, e.g. using intervals $[0, 10], [10, 20], \ldots$. This does not always reflect our intuition, however: by imposing such harsh boundaries, a person who has just turned eleven will not be taken into account in the $[0, 10]$ class, even when she is only at a minimal remove from full membership in that class.

Guided by that observation, many people have suggested alternatives for defining generalized approximation operators, e.g. using axiomatic approaches [18], based on Iwinski-type rough sets [21], in terms of α-cuts [34], level fuzzy sets [17] or fuzzy inclusion measures [15], etc. Some authors (e.g. [30, 34]) explicitly distinguish between rough fuzzy sets (approximations of a fuzzy set in a crisp approximation space) and fuzzy rough sets (approximations of a crisp set in a fuzzy approximation space, i.e., defined by a fuzzy relation R).

A fairly general definition of a fuzzy rough set, absorbing earlier suggestions in the same direction, was given by Radzikowska and Kerre [27]. They paraphrased formulas (3) and (4), which hold in the crisp case, to define the lower and upper approximation of a fuzzy set A in X as the fuzzy sets $R \downarrow A$ and $R \uparrow A$ in X, constructed by means of an implicator $\mathcal{I}$, a t-norm $\mathcal{T}$ and a fuzzy $\mathcal{T}$-equivalence relation R in X,

$$R \downarrow A(y) = \inf_{x \in X} \mathcal{I}(R(x, y), A(x)) \tag{16}$$

$$R \uparrow A(y) = \sup_{x \in X} \mathcal{T}(R(x, y), A(x)) \tag{17}$$

for all y in X. (A_1, A_2) is called a fuzzy rough set (in (X, R)) as soon as there is a fuzzy set A in X such that $R{\downarrow}A = A_1$ and $R{\uparrow}A = A_2$. Formulas (16) and (17) for $R{\downarrow}A$ and $R{\uparrow}A$ can also be interpreted as the degree of inclusion of Ry in A and the degree of overlap of Ry and A respectively, which indicates the semantical link with (1) and (2).

What this definition does not take into account, however, is the fact that if R is a fuzzy $\mathcal{T}$-equivalence relation then it is quite normal that, because of the intermediate degrees of

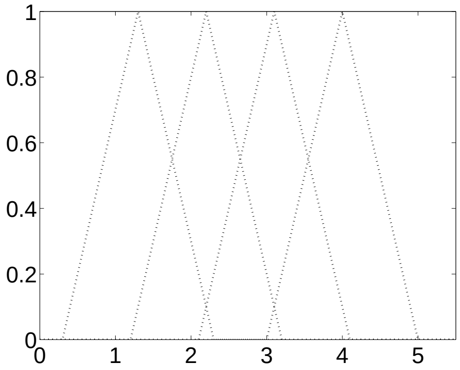


Figure 1: Fuzzy similarity classes

membership, different foresets are not necessarily disjoint. The following example, taken from [9], illustrates this.

Example 3 In applications $\mathcal{T}_{\mathrm{W}}$ is often used as a t-norm because the notion of fuzzy $\mathcal{T}_{\mathrm{W}}$-equivalence relation is dual to that of a pseudo-metric [3]. Let the fuzzy $\mathcal{T}_{\mathrm{W}}$-equivalence relation R in $\mathbb{R}$ be defined by

$$R(x, y) = \max(1 - |x - y|, 0)$$

for all x and y in $\mathbb{R}$. Figure 1 depicts the R-foresets of 1.3, 2.2, 3.1 and 4.

The R-foresets of 3.1 and 4 are clearly different. Still one can easily see that

$$R(3.1, 3.5) = 0.6$$

$$R(4.0, 3.5) = 0.5$$

Since $\mathcal{T}_{\mathrm{W}}(0.6, 0.5) = 0.1$, 3.5 belongs to degree 0.1 to the $\mathcal{T}_{\mathrm{W}}$-intersection of the R-foresets of 3.1 and 4, i.e., these R-foresets are not disjoint.

In other words, the traditional distinction between equivalence and non-equivalence relations is lost when moving on to a fuzzy $\mathcal{T}$-equivalence relation, so it makes sense to exploit the fact that an element can belong to some degree to several R-foresets of *any* fuzzy relation R at the same time. Natural generalizations to the definitions from Section 2.1.2 were therefore proposed in [7, 9].

Definition 4 Let R be a fuzzy relation in X and A a fuzzy set in X.

1. The tight, loose and (usual) lower approximation of A are defined as

   (a) $R{\downarrow}{\downarrow}A(y) = \inf\limits_{z \in X} \mathcal{I}(Rz(y), \inf\limits_{x \in X} \mathcal{I}(Rz(x), A(x)))$

(b) $R{\uparrow}{\downarrow}A(y) = \sup\limits_{z\in X} \mathcal{T}(Rz(y), \inf\limits_{x\in X} \mathcal{I}(Rz(x), A(x)))$

(c) $R{\downarrow}A(y) = \inf\limits_{x\in X} \mathcal{I}(Ry(x), A(x))$

for all y in X.

2. The tight, loose and (usual) upper approximation of A are defined as

(a) $R{\downarrow}{\uparrow}A(y) = \inf\limits_{z\in X} \mathcal{I}(Rz(y), \sup\limits_{x\in X} \mathcal{T}(Rz(x), A(x)))$

(b) $R{\uparrow}{\uparrow}A(y) = \sup\limits_{z\in X} \mathcal{T}(Rz(y), \sup\limits_{x\in X} \mathcal{T}(Rz(x), A(x)))$

(c) $R{\uparrow}A(y) = \sup\limits_{x\in X} \mathcal{T}(Ry(x), A(x))$

for all y in X.

In the next subsection, we investigate the main properties of these alternative approximation operators.

3.2 Properties of Fuzzy Rough Sets

In this section, we will assume that R is a fuzzy tolerance relation in X. Some properties require additional $\mathcal{T}$-transitivity of R; whenever this is the case we mention it explicitly. An overview of the properties discussed in this section is given in Table 5.

3.2.1 Links between the Approximations

Just like in the crisp case, tight and loose approximation operators can be expressed in terms of the usual ones, due to the symmetry of R.

Proposition 5 For every fuzzy set A in X

$$\begin{aligned} R{\downarrow}{\downarrow}A &= R{\downarrow}(R{\downarrow}A) \quad (18)\\ R{\uparrow}{\downarrow}A &= R{\uparrow}(R{\downarrow}A) \quad (19)\\ R{\downarrow}{\uparrow}A &= R{\downarrow}(R{\uparrow}A) \quad (20)\\ R{\uparrow}{\uparrow}A &= R{\uparrow}(R{\uparrow}A) \quad (21)\end{aligned}$$

The monotonicity of the approximations follows easily due to the monotonicity of the fuzzy logical operators involved. This is reflected in the next proposition.

Proposition 6 For every fuzzy set A and B in X

$$A \subseteq B \Rightarrow \begin{cases} R{\downarrow}A \subseteq R{\downarrow}B \\ R{\uparrow}A \subseteq R{\uparrow}B \\ R{\uparrow}{\uparrow}A \subseteq R{\uparrow}{\uparrow}B \\ R{\downarrow}{\uparrow}A \subseteq R{\downarrow}{\uparrow}B \\ R{\uparrow}{\downarrow}A \subseteq R{\uparrow}{\downarrow}B \\ R{\downarrow}{\downarrow}A \subseteq R{\downarrow}{\downarrow}B \end{cases} \tag{22}$$

The following proposition supports the idea of approximating a concept from the lower and the upper side.

Proposition 7 [27] For every fuzzy set A in X

$$R{\downarrow}A \subseteq A \subseteq R{\uparrow}A \tag{23}$$

For the tight and loose approximations, due to Propositions 5, 6 and 7, we can make the following general observations.

Proposition 8 For every fuzzy set A in X

$$R{\downarrow}{\downarrow}A \subseteq R{\downarrow}A \subseteq A \subseteq R{\uparrow}A \subseteq R{\uparrow}{\uparrow}A \tag{24}$$

$$R{\downarrow}A \subseteq R{\uparrow}{\downarrow}A \subseteq R{\uparrow}A \tag{25}$$

$$R{\downarrow}A \subseteq R{\downarrow}{\uparrow}A \subseteq R{\uparrow}A \tag{26}$$

However, the proposition does not give any immediate information about a direct relationship between the loose lower and the tight upper approximation in terms of inclusion, and about how A itself fits in this picture. The following proposition sheds some light on this matter.

Proposition 9 If $\mathcal{T}$ and $\mathcal{I}$ satisfy $\mathcal{T}(x, \mathcal{I}(x, y)) \leq y$ and $y \leq \mathcal{I}(x, \mathcal{T}(x, y))$ for all x and y in $[0, 1]$ then for every fuzzy set A in X

$$R{\uparrow}{\downarrow}A \subseteq A \subseteq R{\downarrow}{\uparrow}A \tag{27}$$

In particular, if $\mathcal{T}$ is a left-continuous t-norm and $\mathcal{I}$ is its residual implicator the property holds [1]. Proposition 9 does not hold in general for other choices of t-norms and implicators as the next example illustrates.

Example 10 Consider the fuzzy $\mathcal{T}$-equivalence relation R on $X = \{a, b\}$ given by

R	a	b
a	1.0	0.2
b	0.2	1.0

and the fuzzy set A in X defined by $A(a) = 1$ and $A(b) = 0.8$. Furthermore let $\mathcal{T} = \mathcal{T}_{\mathrm{M}}$ and $\mathcal{I} = \mathcal{I}_{\mathcal{S}_{\mathrm{M}},\mathcal{N}_s}$. Then $R{\uparrow}A(a) = 1$ and $R{\uparrow}A(b) = 0.8$, hence

$$(R{\downarrow}{\uparrow}A)(a) = \min(\max(0, 1), \max(0.8, 0.8)) = 0.8 \tag{28}$$

which makes it clear that $A \not\subseteq R{\downarrow}{\uparrow}A$.

From all of the above we obtain, for any fuzzy relation R in X,

$$R{\downarrow}{\downarrow}A \subseteq R{\downarrow}A \subseteq R{\uparrow}{\downarrow}A \subseteq A \subseteq R{\downarrow}{\uparrow}A \subseteq R{\uparrow}A \subseteq R{\uparrow}{\uparrow}A \tag{29}$$

provided that $\mathcal{T}$ satisfies the conditions of Proposition 9.

3.2.2 Interaction with Set-Theoretic Operations

The following proposition shows that, given some elementary conditions on the involved connectives, the usual lower and upper approximation are dual w.r.t. fuzzy set complementation.

Proposition 11 [4] If $\mathcal{T}$ is a t-norm, $\mathcal{N}$ an involutive negator and $\mathcal{I}$ the corresponding S-implicator; or, if $\mathcal{T}$ is a left-continuous t-norm, $\mathcal{I}$ its residual implicator and $\mathcal{N}$ defined by $\mathcal{N}(x) = \mathcal{I}(x, 0)$ for x in $[0, 1]$ is an involutive negator, then

$$R{\uparrow}A = co_{\mathcal{N}}(R{\downarrow}(co_{\mathcal{N}}A)) \tag{30}$$

$$R{\downarrow}A = co_{\mathcal{N}}(R{\uparrow}(co_{\mathcal{N}}A)) \tag{31}$$

Combining this result with Proposition 5, it is easy to see that under the same conditions, tight upper and loose lower approximation are dual w.r.t. complementation, as are loose upper and tight lower approximation.

Proposition 12 [27] For any fuzzy sets A and B in X

$$R{\downarrow}(A \cap B) = R{\downarrow}A \cap R{\downarrow}B \tag{32}$$

$$R{\uparrow}(A \cap B) \subseteq R{\uparrow}A \cap R{\uparrow}B \tag{33}$$

$$R{\downarrow}(A \cup B) \subseteq R{\downarrow}A \cup R{\downarrow}B \tag{34}$$

$$R{\uparrow}(A \cup B) = R{\uparrow}A \cup R{\uparrow}B \tag{35}$$

Again by Proposition 5, one can also verify the following equalities

$$R{\downarrow}{\downarrow}(A \cap B) = R{\downarrow}{\downarrow}A \cap R{\downarrow}{\downarrow}B \tag{36}$$

$$R{\uparrow}{\uparrow}(A \cup B) = R{\uparrow}{\uparrow}A \cup R{\uparrow}{\uparrow}B \tag{37}$$

whereas for the remaining interactions, the same inclusions hold as in the crisp case (see Table 2).

3.2.3 Maximal Expansion and Reduction

Taking an upper approximation of A in practice corresponds to expanding A, while a lower approximation is meant to reduce A. However this refining process does not go on forever. The following property says that with the loose lower and the tight upper approximation maximal reduction and expansion are achieved within one approximation.

Proposition 13 [1] If $\mathcal{T}$ is a left-continuous t-norm and $\mathcal{I}$ its residual implicator then for every fuzzy set A in X

$$R{\uparrow}{\downarrow}(R{\uparrow}{\downarrow}A) = R{\uparrow}{\downarrow}A \text{ and } R{\downarrow}{\uparrow}(R{\downarrow}{\uparrow}A) = R{\downarrow}{\uparrow}A \tag{38}$$

To investigate the behaviour of the loose upper and tight lower approximation w.r.t. expansion and reduction, we first establish links with the composition of R with itself. Recall that the composition of fuzzy relations R and S in X is the fuzzy relation $R \circ S$ in X defined by

$$(R \circ S)(x,z) = \sup_{y \in X} \mathcal{T}(R(x,y), S(y,z)) \tag{39}$$

for all x and z in X.

Proposition 14 [9] If $\mathcal{T}$ is a left-continuous t-norm then for every fuzzy set A in X

$$R{\uparrow}{\uparrow}A = (R \circ R){\uparrow}A \tag{40}$$

Proposition 15 [9] If $\mathcal{I}$ is left-continuous in its first component and right-continuous in its second component, and if $\mathcal{T}$ and $\mathcal{I}$ satisfy the shunting principle

$$\mathcal{I}(\mathcal{T}(x,y),z) = \mathcal{I}(x,\mathcal{I}(y,z)) \tag{41}$$

then for every fuzzy set A in X

$$R{\downarrow}{\downarrow}A = (R \circ R){\downarrow}A \tag{42}$$

Note 16 Regarding the restrictions placed on the fuzzy logical operators involved, recall that the shunting principle is satisfied both by a left continuous t-norm and its residual implicator [22] as well as by a t-norm and an S-implicator induced by it [27].

Let us use the following notation, for $n > 1$,

$$R^1 = R \text{ and } R^n = R \circ R^{n-1} \tag{43}$$

From Proposition 14 it follows that taking the upper approximation of a fuzzy set under R n times successively corresponds to taking the upper approximation once under the composed fuzzy relation R^n. Proposition 15 states a similar result for the lower approximation. For the particular case of a fuzzy $\mathcal{T}$-equivalence relation, we have the following important result.

Proposition 17 [27] If R is a fuzzy $\mathcal{T}$-equivalence relation in X then

$$R \circ R = R \tag{44}$$

In other words, using a $\mathcal{T}$-transitive fuzzy relation R, options (1a) and (1c) of Definition 4 coincide, as well as options (2b) and (2c). The following proposition states that under these conditions, they also coincide with (1b), respectively (2a).

Proposition 18 [1, 27] If R is a fuzzy $\mathcal{T}$-equivalence relation in X, $\mathcal{T}$ is a left-continuous t-norm and $\mathcal{I}$ its residual implicator then for every fuzzy set A in X

$$R{\uparrow}{\downarrow}A = R{\downarrow}A \text{ and } R{\downarrow}{\uparrow}A = R{\uparrow}A \tag{45}$$

This means that, using a fuzzy $\mathcal{T}$-equivalence relation to model approximate equality, we will obtain maximal reduction or expansion in one phase, regardless of which of the approximations from Definition 4 is used. As Example 2 already illustrated for the crisp case, when we abandon ($\mathcal{T}$-)transitivity, this behaviour is not always exhibited. In general, when R is not $\mathcal{T}$-transitive and the universe X is finite, it is known that the $\mathcal{T}$-transitive closure of R is given by $R^{|X-1|}$ (assuming $|X| \geq 2$) [19], hence

$$R \circ R^{|X-1|} = R^{|X-1|} \tag{46}$$

In other words with the lower and upper approximation, maximal reduction and expansion will be reached in at most $|X - 1|$ steps, while with the tight lower and the loose upper approximation it can take at most $\lceil |X - 1|/2 \rceil$ steps.

Note 19 The special situation regarding fuzzy $\mathcal{T}$-equivalence relations deserves some further attention. While they are known as the counterpart of equivalence relations, we illustrated in Section 3.1 that their fuzzy similarity classes are not always equal or disjoint; in fact y can belong at the same time to different fuzzy similarity classes to a certain degree. Hence it is not possible, at first sight, to rule out the usefulness of the tight and loose lower and upper approximations introduced in Definition 4. However, careful investigation of the properties of the approximations shows that interplay between suitably chosen fuzzy logical operators and the $\mathcal{T}$-transitivity of the fuzzy relation forces the various approximations to coincide. In the next section we will illustrate that this is not always a desirable property in applications, because it does not allow for gradual expansion or reduction of a fuzzy set by iteratively taking approximations. Omitting the requirement of $\mathcal{T}$-transitivity is precisely the key that allows for a gradual expansion process.

Other undesirable effects of $\mathcal{T}$-transitivity w.r.t. approximate equality were pointed out in [5], [6]. More in particular it is observed there that fuzzy $\mathcal{T}$-equivalence relations can never satisfy the so-called Poincaré paradox. A fuzzy relation R in X is compatible with the Poincaré paradox iff

$$(\exists (x, y, z) \in X^3)(R(x, y) = 1 \wedge R(y, z) = 1 \wedge R(x, z) < 1) \tag{47}$$

This is inspired by Poincaré's [24] experimental observation that a bag of sugar of 10 grammes and a bag of 11 grammes can be perceived as indistinguishable by a human being. The same applies for a bag of 11 grammes w.r.t. a bag of 12 grammes, while the subject is perfectly capable of noting a difference between the bags of 10 and 12 grammes. Now if R is a fuzzy $\mathcal{T}$-equivalence relation, then $R(x, y) = 1$ implies $Rx = Ry$ [10]. Since $Ry(z) = R(y, z) = 1$, also $Rx(z) = R(x, z) = 1$ which is in conflict with $R(x, z) < 1$. The fact that they are not compatible with the Poincaré paradox makes fuzzy $\mathcal{T}$-equivalence relations less suited to model approximate equality. The main underlying cause for this conflict is $\mathcal{T}$-transitivity.

4 Application to query refinement

One of the most common ways to retrieve information from the WWW is keyword based search: the user inputs a query consisting of one or more keywords and the search system returns a list of web documents ranked according to their relevance to the query. The same procedure is often used in e-commerce applications that attempt to relate the user's query to products from the catalogue of some company.

In the basic approach, documents are not returned as search results if they do not contain (one of) the exact keywords of the query. There are various reasons why such an approach might fall short. On one hand there are word mismatch problems: the user knows what he is looking for and he is able to describe it, but the query terms he uses do not exactly correspond to those in the document containing the desired information because of differences in terminology. This problem is even more significant in the context of the WWW than in other, more focussed information retrieval applications, because of the very heterogeneous sources of information expressed in different jargon or even in different natural languages. Note that, on a more general level, a great deal of the Semantic Web efforts are concerned with this problem too, which is reflected by all the attention paid to the construction and the representation of ontologies, allowing agents to communicate with each other by providing a shared and common understanding that reaches across people and application systems (see e.g. [12]). In this paper we rely on a basic kind of ontology, called a thesaurus, which is a term-term relation.

Besides differences in terminology, it is also not uncommon for a user not to be able to describe accurately what he is looking for: the well known "I will know it when I see it" phenomenon. Furthermore, many terms in natural language are ambiguous. For example, a user querying for `java` might be looking for information about either the programming language, the coffee, or the island of Indonesia. To satisfy users who expect search engines to come up with "what they mean and not what they say", it is clear that more sophisticated techniques are needed than a straightforward returning of the documents that contain (one of) the query terms given by the user. One option is query refinement. Since web queries tend to be short—according to [32] they consist of one or two terms on average—we focus on query expansion, i.e. the process of adding related terms to the

Table 5: Properties of lower and upper approximation in a fuzzy approximation space (X, R) (R is a fuzzy tolerance relation).

	Property	Conditions
1.	$R{\uparrow}A = co_{\mathcal{N}}(R{\downarrow}(co_{\mathcal{N}}A))$	
	$R{\downarrow}A = co_{\mathcal{N}}(R{\uparrow}(co_{\mathcal{N}}A))$	$\mathcal{N}$ involutive, $\mathcal{I} = \mathcal{I}_{\mathcal{T},\mathcal{N}}$;
	$R{\downarrow}{\uparrow}A = co_{\mathcal{N}}(R{\uparrow}{\downarrow}(co_{\mathcal{N}}A))$	or, $\mathcal{T}$ left-continuous, $\mathcal{I} = \mathcal{I}_{\mathcal{T}}$
	$R{\uparrow}{\downarrow}A = co_{\mathcal{N}}(R{\downarrow}{\uparrow}(co_{\mathcal{N}}A))$	and $\mathcal{N}(x) = \mathcal{I}(x, 0)$, $\mathcal{N}$ involutive
	$R{\uparrow}{\uparrow}A = co_{\mathcal{N}}(R{\downarrow}{\downarrow}(co_{\mathcal{N}}A))$	(Proposition 11)
	$R{\downarrow}{\downarrow}A = co_{\mathcal{N}}(R{\uparrow}{\uparrow}(co_{\mathcal{N}}A))$	
2.	$R{\downarrow}{\downarrow}A \subseteq R{\downarrow}A \subseteq R{\uparrow}{\downarrow}A \subseteq A$	$\mathcal{T}(x, \mathcal{I}(x, y)) \leq y$ and $y \leq \mathcal{I}(x, \mathcal{T}(x, y))$
	$A \subseteq R{\downarrow}{\uparrow}A \subseteq R{\uparrow}A \subseteq R{\uparrow}{\uparrow}A$	(Proposition 8 and 9)
3.	$A \subseteq B \Rightarrow \begin{cases} R{\downarrow}A \subseteq R{\downarrow}B \\ R{\uparrow}A \subseteq R{\uparrow}B \\ R{\downarrow}{\uparrow}A \subseteq R{\downarrow}{\uparrow}B \\ R{\uparrow}{\downarrow}A \subseteq R{\uparrow}{\downarrow}B \\ R{\uparrow}{\uparrow}A \subseteq R{\uparrow}{\uparrow}B \\ R{\downarrow}{\downarrow}A \subseteq R{\downarrow}{\downarrow}B \end{cases}$	Always (Proposition 6)
4.	$R{\downarrow}(A \cap B) = R{\downarrow}A \cap R{\downarrow}B$	
	$R{\uparrow}(A \cap B) \subseteq R{\uparrow}A \cap R{\uparrow}B$	
	$R{\downarrow}{\uparrow}(A \cap B) \subseteq R{\downarrow}{\uparrow}A \cap R{\downarrow}{\uparrow}B$	Always (Proposition 12)
	$R{\uparrow}{\downarrow}(A \cap B) \subseteq R{\uparrow}{\downarrow}A \cap R{\uparrow}{\downarrow}B$	
	$R{\uparrow}{\uparrow}(A \cap B) \subseteq R{\uparrow}{\uparrow}A \cap R{\uparrow}{\uparrow}B$	
	$R{\downarrow}{\downarrow}(A \cap B) = R{\downarrow}{\downarrow}A \cap R{\downarrow}{\downarrow}B$	
5.	$R{\downarrow}(A \cup B) \supseteq R{\downarrow}A \cup R{\downarrow}B$	
	$R{\uparrow}(A \cup B) = R{\uparrow}A \cup R{\uparrow}B$	
	$R{\downarrow}{\uparrow}(A \cup B) \supseteq R{\downarrow}{\uparrow}A \cup R{\downarrow}{\uparrow}B$	Always (Proposition 12)
	$R{\uparrow}{\downarrow}(A \cup B) \supseteq R{\uparrow}{\downarrow}A \cup R{\uparrow}{\downarrow}B$	
	$R{\uparrow}{\uparrow}(A \cup B) = R{\uparrow}{\uparrow}A \cup R{\uparrow}{\uparrow}B$	
	$R{\downarrow}{\downarrow}(A \cup B) \supseteq R{\downarrow}{\downarrow}A \cup R{\downarrow}{\downarrow}B$	
6.	$R{\downarrow}{\uparrow}(R{\downarrow}{\uparrow}A) = R{\downarrow}{\uparrow}A$	$\mathcal{T}$ left-continuous, $\mathcal{I} = \mathcal{I}_{\mathcal{T}}$
	$R{\uparrow}{\downarrow}(R{\uparrow}{\downarrow}A) = R{\uparrow}{\downarrow}A$	(Proposition 13)
	$R{\uparrow}{\downarrow}A = R{\downarrow}{\downarrow}A = R{\downarrow}A$	R a fuzzy $\mathcal{T}$-equivalence relation in X,
	$R{\downarrow}{\uparrow}A = R{\uparrow}{\uparrow}A = R{\uparrow}A$	$\mathcal{T}$ left-continuous, $\mathcal{I} = \mathcal{I}_{\mathcal{T}}$ (Proposition 17 and 18)

query, and demonstrate how the approximation operators from the previous section may assist us in this task.

4.1 Related Work

Query refinement has found its way to popular web search engines, and is even becoming one of those features in which search engines aim to differentiate in their attempts to create their own identity. Simultaneously with search results, Yahoo![1] shows a list of clickable expanded queries in an "Also Try" option under the search box. These queries are derived from logs containing queries performed earlier by others. Google Suggest[2] also uses data about the overall popularity of various searches to help rank the refinements it offers, but unlike the other search engines, the suggestions pop up in the search box while you type, i.e., before you search. Ask.com[3] provides a zoom feature, allowing users to narrow or broaden the field of search results, as well as view results for related concepts.

Query expansion goes back a long way before the existence of the WWW though. Over the last decades several important techniques have been established. The main idea underlying all of them, is to extend the query with words related to the query terms. One option is to use an available thesaurus such as WordNet[4], expanding the query by adding synonyms [31]. Related terms can also be automatically discovered from the searchable documents though, taking into account statistical information such as co-occurrences of words in documents or in fragments of documents. The more terms co-occur, the more they are assumed to be related. In [32] several of these approaches are discussed and compared. In global document analysis, the whole corpus of searchable documents is preprocessed and transformed into an automatically generated thesaurus. Local document analysis on the other hand only considers the top ranked documents for the initial query. In its most naive form, terms that appear most frequently in these top ranked documents are added to the query. Local document analysis is referred to as a pseudo-relevance feedback approach, because it tacitly assumes that the highest ranked documents are indeed relevant to the query. A true relevance feedback approach takes into account the documents marked as relevant by the user. Finally, in [2], correlations between terms are computed based on their co-occurrences in query logs instead of in documents.

Once the relationship between terms is known, either through a lexical aid such as WordNet, or automatically generated from statistical information, the original query can be expanded in various ways. The straightforward way is to extend the query with all the words that are related to at least one of the query terms. Intuitively, this corresponds to taking the upper approximation of the query. Indeed, a thesaurus characterizes an approximation space in which the query, which is a set of terms, can be approximated from the upper (and the lower) side. By definition, the upper approximation will add a term to the query as soon as it is related to *one* of the words already in the query. This link

[1] http://search.yahoo.com/

[2] http://labs.google.com/suggest/

[3] http://www.ask.com/

[4] http://wordnet.princeton.edu/

between query expansion and rough set theory has been established in [28], even involving fuzzy logical representations of the term-term relations and the queries.

In [31], it is pointed out, however, that such an approach requires sense resolution of ambiguous words. Indeed, the precision of retrieved documents is likely to decrease when expanding a query such as *java, travel* with the term *applet*. Even though this term is highly related to *java* as a programming language, it has little or nothing to do with the intended meaning of *java* in this particular query, namely the island. An option to automate sense disambiguation is to only add a term when it is related to at least two words of the original query; experimental results are however unsatisfactory [31].

In [2], the most popular sense gets preference. For example, if the majority of users use windows to search for information about the Microsoft product, the term windows has much stronger correlations with terms such as Microsoft, OS and software, rather than with terms such as decorate, door and house. The approaches currently taken by Yahoo! and Google Suggest seem to be in line with this principle. Note, however, that these search engines do not apply query expansion automatically but leave the final decision up to the user.

In [26], a virtual term is created to represent the general concept of the query. Terms are selected for expansion based on their similarity to this virtual term. In [32], candidate expansion terms are ranked based on their co-occurrence with all query terms in the top ranked documents.

4.2 Finding the Right Balance: Query Expansion using the Tight Upper Approximation

The approach discussed here, first introduced in [8] and taken up also in [29], differs from all techniques mentioned above, and takes into account the lower approximation as well. The lower approximation will only retain a term in the query if *all* the words that it is related too are also in the query. It is obvious that the lower approximation will easily result in the empty query, hence in practice it is often too strict for query refinement. On the other hand, it is not hard to imagine cases where the upper approximation is too flexible as a query expansion technique, resulting not only in an explosion of the query, but possibly even worse, in the addition of non relevant terms due to the ambiguous nature of one or more of the query words. This is due to the fact that the upper approximation expands each of the query words individually but disregards the query as a whole.

As will become clear in the next sections, we go further than the expansion of individual query terms, but we do not go as far as restricting ourselves to words that are related to at least two or preferably all terms of the initial query. Instead, we follow an approach where terms can be added as long as they are not strongly related to words that have nothing to do with the query at all. As such, this approach contributes to the problem of automatic query disambiguation in search engines [13].

We suggest to combine the flexibility of the upper approximation with the strictness of the lower approximation by applying them successively. As such, first we expand the

Table 6: Millions of web pages found by Google.

# documents	mac	computer	apple	fruit	pie	recipe	store	emulator	hardware
mac	$\underline{114}$	18.3	14.9	1.03	0.869	0.899	15.8	0.672	15.1
computer		$\underline{375}$	15.6	3.76	2.22	3.72	29.5	1.17	26.9
apple			$\underline{93.4}$	5.42	3.81	4.59	14.3	0.401	17.8
fruit				$\underline{35.4}$	2.32	4.08	7.63	0.047	1.63
pie					$\underline{20.4}$	4.21	3.74	0.030	1.2
recipe						$\underline{31.5}$	6.22	0.035	1.69
store							$\underline{312}$	0.472	24.9
emulator								$\underline{4.95}$	1.05
hardware									$\underline{178}$

query by adding all the terms that are known to be related to at least one of the query words. Next we reduce the expanded query by taking its lower approximation, thereby pruning away all previously added terms that are suspected to be irrelevant for the query. The pruning strategy targets those terms that are strongly related to words that do not belong to the expanded query.

Our technique can be used both with a crisp thesaurus in which terms are related or not, as with a graded thesaurus in which terms are related to some degree. Furthermore it can be applied for weighted as well as for non-weighted queries. Whenever the user does not want to go through the effort of assigning individual weights to query terms, they are all given the highest weight by default. When a graded thesaurus is used, our query expansion approach turns the original query automatically into a weighted query. The original user-chosen terms maintain their highest weight, and new terms are added with weights that do not only reflect the strength of the relationship with the original individual query terms as can be read from the thesaurus, but also take into account their relevance to the query as a whole. To be able to deal with graded thesauri and weighted queries and apply the machinery of fuzzy rough sets, we represent the thesaurus as a fuzzy relation and the query as a fuzzy set.

4.2.1 Thesaurus Construction

Table 7 shows a small sample fuzzy thesaurus R. In constructing it, we did not use any direct human expert knowledge regarding the semantics of the terms involved, but we relied on the number of web pages found by a search engine for each pair of terms, as shown in Table 6.

On the WWW there is a strong bias towards computer science related terms, hence the absolute number of web pages containing both term t_1 and t_2 cannot be used directly to express the strength of the relationship between t_1 and t_2. To level out the difference, we

Table 7: Graded thesaurus.

R	mac	computer	apple	fruit	pie	recipe	store	emulator	hardware
mac	1.00	0.89	0.89	0.00	0.01	0.00	0.75	0.83	0.66
computer		1.00	0.94	0.44	0.44	0.56	0.25	1.00	0.83
apple			1.00	0.83	0.99	0.83	0.83	0.25	0.99
fruit				1.00	0.44	0.66	1.00	0.00	0.03
pie					1.00	1.00	0.97	0.00	0.06
recipe						1.00	1.00	0.00	0.03
store							1.00	0.34	0.75
emulator								1.00	1.00
hardware									1.00

used the following measure

$$\frac{|D_{t_1} \cap D_{t_2}|}{\min(|D_{t_1}|, |D_{t_2}|)} \tag{48}$$

where D_{t_1} and D_{t_2} denote the sets of web pages that contain term t_1, respectively term t_2. Finally, we normalized the result using the S-function $S(.; 0.03, 0.20)$ (cfr. Figure 2), giving rise to the fuzzy thesaurus R of Figure 7.

The fuzzy relation R characterizing the approximation space is a fuzzy tolerance relation that is not $\mathcal{T}$-transitive for any choice of $\mathcal{T}$. To see, this note that

$$\mathcal{T}(R(\text{pie,recipe}), R(\text{recipe,fruit})) = \mathcal{T}(1, 0.66) = 0.66 > 0.44 = \mathcal{T}(\text{pie,fruit}) \tag{49}$$

For comparison purposes, we also constructed a $\mathcal{T}_W$-transitive fuzzy thesaurus by taking the $\mathcal{T}_W$-transitive closure $R^{|X-1|}$ of R, i.e., the smallest $\mathcal{T}_W$-transitive fuzzy relation in which R is included. This thesaurus is shown in Table 8. In our running example, to compute upper and lower approximations, we will keep on using the t-norm $\mathcal{T}_\mathrm{W}$ as well as its residual implicator $\mathcal{I}_{\mathcal{T}_\mathrm{W}}$. Finally we constructed a crisp (i.e., non-graded) thesaurus by taking the 0.5-level of R, defined as

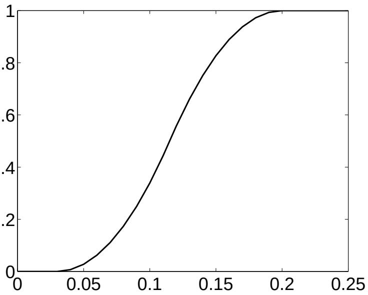
$$(x, y) \in R_{.5} \text{ iff } R(x, y) \geq 0.5 \tag{50}$$

for all x and y in X. In other words, in the crisp thesaurus, depicted in Table 9, two terms are related if and only if the strength of their relationship in the graded thesaurus R of Table 7 is at least 0.5.

It can be easily verified that $R_{.5}$ is not transitive. For example, *fruit* is related to *store* and *store* is related to *hardware*, but *fruit* is not related to *hardware.* For comparison purposes, in the remainder, we also include the transitive closure $(R_{.5})^8$.

4.2.2 Query Refinement

We consider the query



$$S(x;\alpha,\gamma) = \begin{cases} 0, & \text{if } x \leq \alpha \\ \frac{2(x-\alpha)^2}{(\gamma-\alpha)^2}, & \text{if } \alpha \leq x \leq (\alpha+\gamma)/2 \\ 1 - \frac{2(x-\gamma)^2}{(\gamma-\alpha)^2}, & \text{if } (\alpha+\gamma)/2 \leq x \leq \gamma \\ 1, & \text{if } \gamma \leq x \end{cases}$$

Figure 2: S-function; x, α, and γ in $\mathbb{R}$, $\alpha < \gamma$

Table 8: Transitive closure of graded thesaurus.

R^8	mac	computer	apple	fruit	pie	recipe	store	emulator	hardware
mac	1.00	0.89	0.89	0.88	0.88	0.88	0.88	0.89	0.89
computer		1.00	0.99	0.99	0.99	0.99	0.99	1.00	1.00
apple			1.00	0.99	0.99	0.99	0.99	0.99	0.99
fruit				1.00	1.00	1.00	1.00	0.99	0.99
pie					1.00	1.00	1.00	0.99	0.99
recipe						1.00	1.00	0.99	0.99
store							1.00	0.99	0.99
emulator								1.00	1.00
hardware									1.00

Table 9: Crisp thesaurus.

$R_{.5}$	mac	computer	apple	fruit	pie	recipe	store	emulator	hardware
mac	1	1	1	0	0	0	1	1	1
computer		1	1	0	0	1	0	1	1
apple			1	1	1	1	1	0	1
fruit				1	0	1	1	0	0
pie					1	1	1	0	0
recipe						1	1	0	0
store							1	0	1
emulator								1	1
hardware									1

Table 10: Upper approximation based query expansion with graded thesaurus.

	A	$R{\uparrow}A$	$R{\uparrow}(R{\uparrow}A)$	$R^8{\uparrow}A$
mac	0.00	0.89	0.89	0.89
computer	0.00	0.94	0.94	0.99
apple	1.00	1.00	1.00	1.00
fruit	0.00	0.83	1.00	1.00
pie	1.00	1.00	1.00	1.00
recipe	1.00	1.00	1.00	1.00
store	0.00	1.00	1.00	1.00
emulator	0.00	0.25	0.99	0.99
hardware	0.00	0.99	0.99	0.99

apple, pie, recipe

as shown in the second column in Table 10. The intended meaning of the ambiguous word *apple*, which can refer both to a piece of fruit and to a computer company, is clear in this query.

The disadvantage of using a $\mathcal{T}$-transitive fuzzy thesaurus becomes apparent when we compute the upper approximation $R^8{\uparrow}A$, shown in the last column. All the terms are added with high degrees, even though terms like *mac* and *computer* have nothing to do with the semantics of the original query. This process can be slowed down a little bit by using the non $\mathcal{T}$-transitive fuzzy thesaurus and computing $R{\uparrow}A$ which allows for some gradual refinement. However an irrelevant term such as *emulator* shows up to a high degree in the second iteration, i.e. when computing $R{\uparrow}(R{\uparrow}A)$. The problem is even more prominent when using a crisp thesaurus as shown in Table 11.

It is important to point out that under our assumptions

$$A \subseteq R{\downarrow}{\uparrow}A \subseteq R{\uparrow}A \tag{51}$$

Table 11: Upper approximation based query expansion with crisp thesaurus.

	A	$R_{.5}{\uparrow}A$	$R_{.5}{\uparrow}(R_{.5}{\uparrow}A)$	$(R_{.5})^8{\uparrow}A$
mac	0	1	1	1
computer	0	1	1	1
apple	1	1	1	1
fruit	0	1	1	1
pie	1	1	1	1
recipe	1	1	1	1
store	0	1	1	1
emulator	0	0	1	1
hardware	0	1	1	1

always holds, guaranteeing that the tight upper approximation indeed leads to an expansion of the query—none of the original terms are lost—and at the same time is a pruned version of the upper approximation. When R is a fuzzy $\mathcal{T}$-equivalence relation, the upper approximation and the tight upper approximation coincide (see Table 5). However, as we show below, this is not necessarily the case when R is not $\mathcal{T}$-transitive.

The main problem with the query expansion process described in the previous section, even if it is gradual, is a fast growth of the number of less relevant or irrelevant keywords that are automatically added. This effect is caused by the use of a flexible definition of the upper approximation in which a term is added to a query as soon as it is related to one of its keywords. However, using the tight upper approximation a term y will only be added to a query A if all the terms that are related to y are also related to at least one keyword of the query. First the usual upper approximation of the query is computed, but then it is stripped down by omitting all terms that are also related to other terms not belonging to this upper approximation. In this way terms that are sufficiently relevant, hence related to most keywords in A, will form a more or less closed context with few or no links outside, while a term related to only one of the keywords in A in general also has many links to other terms outside $R{\uparrow}A$ and hence is omitted by taking the lower approximation.

The last column of Table 12 shows that the tight upper approximation is different from and performs clearly better than the traditional upper approximation for our purpose of web query expansion: irrelevant words such as *mac*, *computer* and *hardware* are still added to the query, but to a significantly lower degree. The difference becomes even more noticable when using a crisp thesaurus as illustrated in Table 13.

Table 12: Comparison of upper and tight upper approximation based query expansion with graded thesaurus.

	A	$R{\uparrow}A$	$R^8{\uparrow}A$	$R{\downarrow}{\uparrow}A$
mac	0.00	0.89	0.89	0.42
computer	0.00	0.94	0.99	0.25
apple	1.00	1.00	1.00	1.00
fruit	0.00	0.83	1.00	0.83
pie	1.00	1.00	1.00	1.00
recipe	1.00	1.00	1.00	1.00
store	0.00	1.00	1.00	0.83
emulator	0.00	0.25	0.99	0.25
hardware	0.00	0.99	0.99	0.25

Table 13: Comparison of upper and tight upper approximation based query expansion with crisp thesaurus.

	A	$R_{.5}{\uparrow}A$	$(R_{.5})^8{\uparrow}A$	$R_{.5}{\downarrow}{\uparrow}A$
mac	0	1	1	0
computer	0	1	1	0
apple	1	1	1	1
fruit	0	1	1	1
pie	1	1	1	1
recipe	1	1	1	1
store	0	1	1	1
emulator	0	0	1	0
hardware	0	1	1	0

5 Conclusion

Acknowledgment

Chris Cornelis would like to thank the Research Foundation–Flanders for funding his research.

References

[1] U. Bodenhofer, A unified framework of opening and closure operators with respect to arbitrary fuzzy relations, Soft Computing **7**, p. 220–227, 2003.

[2] H. Cui, J.R. Wen, J.Y. Nie, W.Y. Ma, Probabilistic query expansion using query logs, Proceedings of WWW2002 (the 11th International World Wide Web Conference), ACM Press, p. 325–332, 2002.

[3] B. De Baets, R. Mesiar, Pseudo-metrics and $\mathcal{T}$-equivalences, The Journal of Fuzzy Mathematics **5**, p. 471–481, 1997.

[4] M. De Cock, A thorough study of linguistic modifiers in fuzzy set theory, Ph.D. thesis (in Dutch), Ghent University, 2002.

[5] M. De Cock, E.E. Kerre, On (un)suitable fuzzy relations to model approximate equality, Fuzzy Sets and Systems **133**(2), p. 137–153, 2003.

[6] M. De Cock, E.E. Kerre, Why fuzzy $\mathcal{T}$-equivalence relations do not resolve the Poincaré paradox, and related issues, Fuzzy Sets and Systems **133**(2), p. 181–192, 2003.

[7] M. De Cock, C. Cornelis, E.E. Kerre, Fuzzy rough sets: beyond the obvious, Proceedings of FUZZ-IEEE2004 (2004 IEEE International Conference on Fuzzy Systems), Volume 1, p. 103-108, 2004.

[8] M. De Cock, C. Cornelis, Fuzzy rough set based web query expansion, Proceedings of Rough Sets and Soft Computing in Intelligent Agent and Web Technology, International Workshop at WIIAT2005 (2005 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology), p. 9–16, 2005.

[9] M. De Cock, C. Cornelis, E.E. Kerre, Fuzzy rough sets: the forgotten step, Special Issue on Extensions to Type-1 Fuzzy Sets (B. John, E. Garibaldi, eds.), IEEE Transactions on Fuzzy Systems, in press.

[10] D. Dubois and H. Prade, Rough fuzzy sets and fuzzy rough sets, International Journal of General Systems **17**, p. 91–209, 1990.

[11] L. Fariñas del Cerro, H. Prade, Rough sets, twofold fuzzy sets and modal logic—Fuzziness in indiscernibility and partial information, The Mathematics of Fuzzy Systems (A. Di Nola, A.G.S. Ventre, eds.), Verlag TUV Rheinland, Köln, p. 103–120, 1986.

[12] D. Fensel, F. van Harmelen, I. Horrocks, D.L. McGuinness, P.F. Patel-Schneider, OIL: an ontology infrastructure for the Semantic Web, IEEE Intelligent Systems **16**, p. 38–45, 2001.

[13] A. Fischer, What's it going to take to beat Google?, Search Engine Watch, June 13, 2003.

[14] T.B. Iwinski, Algebraic approach to rough sets, Bulletin of the Polish Academy of Science and Mathematics **35**, p. 673–683, 1987.

[15] L.I. Kuncheva, Fuzzy rough sets: application to feature selection, Fuzzy Sets and Systems **51**, p. 147-153, 1992.

[16] T.Y. Lin, Topological and fuzzy rough sets, Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory (R. Slowinski, ed.), Kluwer Academic Publishers, Boston, p. 287–304, 1992.

[17] W.N. Liu, J.T. Yao, Y.Y. Yao, Rough approximations under level fuzzy sets, Lecture Notes in Artificial Intelligence **3066**, p. 78-83, 2004.

[18] N.N. Morsi, M.M. Yakout, Axiomatics for fuzzy rough sets, Fuzzy sets and Systems **100**(1–3), p. 327–342, 1998.

[19] H. Naessens, H. De Meyer, B. De Baets, Algorithms for the computation of $\mathcal{T}$-transitive closures, IEEE Transactions on Fuzzy Systems **10**(4), p. 541–551, 2002.

[20] A. Nakamura, Fuzzy rough sets, Note on Multiple-Valued Logic in Japan **9**, p. 1–8, 1988.

[21] S. Nanda, S. Majumdar, Fuzzy rough sets, Fuzzy Sets and Systems **45**, p. 157–160, 1992.

[22] V. Novák, I. Perfilieva, J. Močkoř, Mathematical principles of fuzzy logic, Kluwer Academic Publishers, 1999.

[23] Z. Pawlak, Rough sets, International Journal of Computer and Information Sciences **11**(5), p. 341–356, 1982.

[24] H. Poincaré, La science et l'hypothèse, Flammarion, Paris, 1902.

[25] J.A. Pomykala, Approximation operations in approximation space, Bulletin of the Polish Academy of Sciences, Mathematics **35**, p. 653–662, 1987.

[26] Y. Qui, H. Frei, Concept based query expansion, Proceedings of ACM SIGIR 1993 (16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval), p. 160–169, 1993.

[27] A.M. Radzikowska, E.E. Kerre, A comparative study of fuzzy rough sets, Fuzzy Sets and Systems **126**, p. 137–156, 2002.

[28] P. Srinivasan, M.E. Ruiz, D.H. Kraft, J. Chen, Vocabulary mining for information retrieval: rough Sets and fuzzy Sets, Information Processing and Management **37**, p. 15–38, 2001.

[29] S. Tenreiro de Magalhães, L. Duarte dos Santos, L. Amaral, Getting the knowledge to the agent the rough sets approach, Proceedings of 8th International Conference on Current Research Information Systems, p. 155–166, 2006.

[30] H. Thiele, Fuzzy rough sets versus rough fuzzy sets—an interpretation and a comparative study using concepts of modal logic, Technical Report ISSN 1433-3325, University of Dortmund, 1998.

[31] E.M. Voorhees, Query expansion using lexical-semantic relations, Proceedings of ACM SIGIR 1994 (17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval), p. 61–69, 1994.

[32] J. Xu, W.B. Croft, Query expansion using local and global document analysis, Proceedings of ACM SIGIR 1996 (19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval), p. 4–11, 1996.

[33] Y.Y. Yao, Two views of the theory of rough sets in finite universes, International Journal of Approximate Reasoning **15**(4), p. 291–317, 1996.

[34] Y.Y. Yao, Combination of rough and fuzzy sets based on alpha-level sets, Rough Sets and Data Mining: Analysis for Imprecise Data (T.Y. Lim, N. Cercone, eds.), Kluwer Academic Publishers, Boston, p. 301–321, 1997.

[35] L.A. Zadeh, Fuzzy sets, Information and Control **8**, 338–353, 1965.