Weather Prediction Using Case-Based Reasoning and Fuzzy Set Theory

by

Bjarne Kristian Hansen

A Thesis Submitted to the Faculty of Computer Science

in Partial Fulfillment of the Requirements for the Degree of

MASTER OF COMPUTER SCIENCE

Major Subject: Computer Science

APPROVED:

____

Denis Riordan, Supervisor

____

Mohammed El-Hawary

____

Qigang Gao

DALHOUSIE UNIVERSITY - DALTECH

Halifax, Nova Scotia 2000

DALTECH LIBRARY

“AUTHORITY TO DISTRIBUTE MANUSCRIPT THESIS”

TITLE:
Weather Prediction Using Case-Based Reasoning and Fuzzy Set Theory

The above library may make available or authorize another library to make available individual photo/microfilm copies of this thesis without restrictions.

Full Name of Author: Bjarne Kristian Hansen

Signature of Author: ______________________________

Date: May 17, 2000

To Diane.

TABLE OF CONTENTS

List of Figures

List of Abbreviations and Symbols

Abbreviation	Meaning
AI	artificial intelligence
CBR	case-based reasoning
DBMS	database management systems
DSS	decision support system
EC	Environment Canada
FOH	Frequency of Hits
FAR	False Alarm Ratio
MOS	model output statistics (NWP + climatology + statistics)
MSC	Meteorological Service of Canada (part of Environment Canada)
NWP	numerical weather prediction
POD	Probability of Detection
TAF	Terminal Aerodrome Forecast
VFR	Visual Flight Rules
WIND-1	*Weather Is Not Discrete* - *Version 1*

Symbol	Meaning
μ	degree of membership in a fuzzy set, $0.0 \leq \mu(x) \leq 1.0$

Acknowledgments

Thanks to Diane Ouellet, my better half.

Thanks to my parents, Arthur Hansen and Elisabeth Hansen.

Thanks to all my colleagues in the Meteorological Service of Canada, part of Environment Canada, for their guidance and support during the development of this thesis. Special thanks to Jim Abraham, Bill Appleby, Réal Daigle, Allan MacAfee, Ken Macdonald, Jim Murtha, Martha McCulloch, George Parkes, Bruce Whiffen, and Laurie Wilson. Thanks to the Meteorological Service of Canada for the data and computer facilities used to conduct this research, and for paying for some of the costs.

Thanks to my supervisor Dr. Denis Riordan for showing me how case-based reasoning works and challenging me to "think about the data."

Thanks to my teacher Dr. Mo El-Hawary for showing me how fuzzy logic works and advising me to "beware of imposters in the temple" (think about the source).

Thanks to my teacher Dr. Qigang Gao for showing me how computer vision works and encouraging me to build a foundation for using computer vision with this thesis.

Thanks to Dr. David Aha for recommending relevant articles about case-based reasoning and for suggesting novel research directions.

Thanks to the University of Washington Press for letting me use two figures from *The Essence of Chaos* (Lorenz 1993). These figures clearly illustrate how *chaotic* means *sensitively dependent on initial conditions*. (These figures appear here in Figure 5.)

Thanks to Dr. Agnar Aamodt for letting me use a figure from *Case-based reasoning: Foundational issues, methodological variations, and system approaches* (Aamodt and Plaza 1994). This figure illustrates a frequently referred-to case-based reasoning cycle: retrieve, reuse, revise, retain (This figure appears here in Figure 2.)

Abstract

A fuzzy logic based methodology for knowledge acquisition is developed and used for retrieval of temporal cases in a case-based reasoning system. The methodology is used to acquire knowledge about what salient features of continuous-vector, unique temporal cases indicate significant similarity between cases. Such knowledge is encoded in a similarity-measuring function and thereby used to retrieve *k* nearest neighbors (*k*-nn) from a large database. Predictions for the present case are made from a weighted median of the outcomes of analogous past cases (i.e., the *k*-nn, or the analog ensemble). Past cases are weighted according to their degree of similarity to the present case.

Fuzzy logic imparts to case-based reasoning the perceptiveness and case-discriminating ability of a domain expert. The fuzzy *k*-nn technique retrieves similar cases by emulating a domain expert who understands and interprets similar cases. The main contribution of fuzzy logic to case-based reasoning (CBR) is that it enables us to use common words to directly acquire domain knowledge about feature salience. This knowledge enables us to retrieve a few most similar cases from a large temporal database, which in turn helps us to avoid the problems of case adaptation and case authoring.

Such a fuzzy *k*-nn weather prediction system can improve the technique of persistence climatology (PC) by achieving direct, efficient, expert-like comparison of past and present weather cases. PC is an analog forecasting technique that is widely recognized as a formidable benchmark for short-range weather prediction. Previous PC systems have had two built-in constraints: they represented cases in terms of the memberships of their attributes in predefined categories and they referred to a preselected combination of attributes (i.e., cases defined and selected before receiving the precise and numerous details of present cases). The proposed fuzzy *k*-nn system compares past and present cases directly and precisely in terms of their numerous salient attributes. The fuzzy *k*-nn method is not tied to specific categories, nor is it constrained to using only a specific limited set of predictors. Such a system for making airport weather predictions will let us tap many, large, unused archives of airport weather observations, ready repositories of temporal cases. This will help to make airport weather predictions more accurate, which will make air travel safer and make airlines more profitable.

Accordingly, a fuzzy *k*-nn based prediction system, called WIND-1, is proposed, implemented, and tested. Its unique component is an expertly-tuned fuzzy *k*-nn algorithm with a temporal dimension. It is tested with the problem of producing 6-hour predictions of cloud ceiling and visibility at an airport, given a database of over 300,000 consecutive hourly airport weather observations (36 years of record). Its prediction accuracy is measured with standard meteorological statistics and compared to a benchmark prediction technique, persistence. In realistic simulations, WIND-1 is significantly more accurate. WIND-1 produces forecasts at the rate of about one per minute.

1. Introduction

Fuzzy set theory based methods enable case-based reasoning (CBR) systems developers to impart the perceptiveness and case-discriminating ability of a domain expert to CBR. [1] Our goal is to develop a technique that will increase the usefulness of fuzzy methods for retrieval of similar cases. We deal with temporal cases in which the attributes are continuous variables, cases which are described by spatiotemporal vectors.

We attend to the problem of how to increase the effectiveness of a basic weather prediction technique that is referred to in meteorology as *analog forecasting*—a meteorological form of CBR. Analog forecasting makes predictions for a present weather situation based on the outcomes of similar past weather situations.

Weather prediction presents special challenges for CBR. Weather is continuous, data-intensive, multidimensional, dynamic and chaotic. These five properties make weather prediction a formidable proving ground for any CBR prediction system that depends on searching for similar sequences. Searching for similar sequences is a problem which occurs in diverse applications, such as stock market prediction (Rafiei 1999, and Xia 1997), plagiarism detection (Shivakumar and Garcia-Molina 1995), forest fire prediction (Rougegrez 1993), and protein and DNA sequencing (Pearson and Lipman 1988). So, an effective basic technique for finding similar sequences has potentially wider applicability than for just weather prediction.

Our survey of the literature about the problem of how to determine similarity and about the nearest neighbors technique will be limited. A huge amount of such literature already exists. The problem of how to determine similarity interests researchers from numerous disciplines. Many papers and several books have been written on the subject of nearest neighbors techniques. [2] Most of this literature focuses narrowly on the particular discipline it stems from or the particular application it deals with.

---

[1] The opening statement is the *thesis* of this thesis. Specific support for this statement can be found in a number of articles that we review in Section 2.4 on page 65 (namely: Bonissone and Ayub 1992; Bonissone and Cheetham 1997; Cheetham and Graf 1997; Göös et al. 1999; Hansen and Riordan 1998; Lefley and Austin 1997; Main et al. 1996; Tobin et al. 1998; Weber-Lee et al. 1995; Winder et al. 1997).

[2] The Interdisciplinary Workshop On Similarity And Categorisation (SimCat 97), held at the University of Edinburgh in 1997, was "a gathering of researchers addressing similarity and categorisation from a wide range of disciplines, including: artificial intelligence, machine learning, case-based reasoning, psychology, philosophy, linguistics, statistics, semiotics, music, [and] design theory." (Description of workshop downloaded from http://www.dai.ed.ac.uk/conferences/simcat April 18, 2000)

A book that reprints 51 papers by different researchers is: Dasarthy, B. V. (ed.) 1991, Nearest Neighbor Pattern Classification Techniques, IEEE Computer Society Press, Los Alamitos, CA. All the

We survey literature where the interests of fuzzy logic and CBR intersect (a relatively small and growing subset), and we describe in detail a unique application for weather prediction that uses a combination of fuzzy logic and CBR.

Based on our previous success with weather prediction using CBR and fuzzy logic (Hansen and Riordan 1998), we hypothesize as follows:

1.1 Hypothesis

> *Querying a large database of weather observations for past weather cases similar to a present case using a fuzzy k-nearest neighbors algorithm that is designed and tuned with the help of a weather forecasting expert can increase the accuracy of predictions of cloud ceiling and visibility at an airport.*

1.2 Thesis structure

In the rest of this chapter, we briefly introduce the three subjects of the thesis title: CBR, fuzzy logic, and weather prediction. We focus on how each subject relates to *retrieval of similar cases*. In Section 1.3, we explain how CBR depends on retrieval of similar cases and explain how the applicability of CBR is hindered by the problems of "case adaptation" and "case authoring." In Section 1.4, we explain how fuzzy logic enables retrieval of similar cases. In section 1.5, we introduce the airport weather prediction problem addressed in this thesis, describe the state of the art of artificial intelligence in weather prediction, and explain how a well-known weather prediction technique known as "analog forecasting," which is a meteorological form of CBR, depends on retrieval of similar cases.

In Chapter 2, we survey the literature to focusing on how using a fuzzy k-nearest neighbors based technique for retrieval of similar cases, designed and tuned with the help of domain expert, can help us to exploit large databases of cases and available domain knowledge about similarity, and can help us to avoid difficulties of case adaptation and case authoring. We describe the main resources for CBR, review how fuzzy logic applies to CBR, provide a foundation for the fuzzy k-nearest neighbors (fuzzy *k*-nn) technique, review a number of CBR

---

pattern classification deals with static images, or multi-dimensional cases. There is no explicit coverage of time dimension, or prediction. However, the preface does say that "NN concepts are being applied in new environments outside traditional pattern recognition."

applications that exemplify the fuzzy *k*-nn technique, and review weather prediction papers that use CBR and fuzzy logic.

In Chapter 3, we describe our unique system for fuzzy *k*-nn based weather prediction. In Chapter 4, we describe a set of experiments to test the effectiveness of the system and presents the results. In Chapter 5, we present our conclusions and describe future possible directions for this research.

1.3 Case-based reasoning

In this section, we give a general introduction to CBR. We condense some frequently-quoted articles about CBR. In Chapter 2, we survey articles focusing on the overlaps of CBR, fuzzy logic, and weather prediction.

Case-based reasoning is a method for solving problems by remembering previous similar situations and reusing information and knowledge about that situation (Kolodner 1993; and Leake 1996). The original, basic idea is simple:

> A case-based reasoner solves new problems by adapting solutions that were used to solve old problems. (Riesbeck and Schank 1989)

CBR is very effective in situations "where the acquisition of the case-base and the determination of the features is straightforward compared with the task of developing the reasoning mechanism." (Cunningham and Bonzano 1999).

A classic flowchart for case-based reasoning is shown in Figure 1. The flowchart is basically the same as that of (Riesbeck and Schank 1989). We have reformatted their flowchart slightly to highlight knowledge-acquisition problems that continue to challenge CBR research.

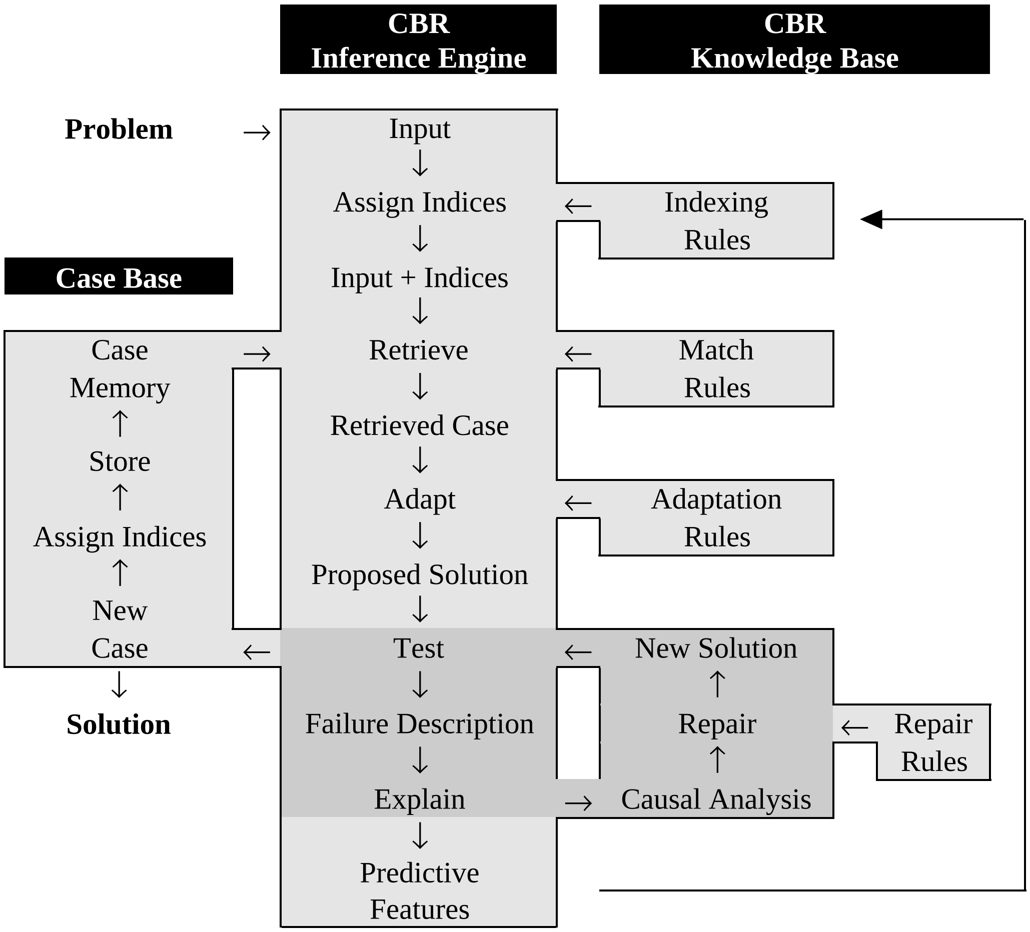


Figure 1. Classic case-based reasoning flowchart. This flowchart, conceptually the same as that of (Riesbeck and Schank 1989), shows how knowledge acquisition is a fundamental challenge for CBR system developers. Developers must acquire knowledge about how to index and match cases, how to adapt cases into solutions, and how to repair failed solutions.

Based on an extensive survey of CBR, Aamodt and Plaza (1994) describe CBR as a four-step process:

- *Retrieve* the most similar case or cases.
- *Reuse* the information and knowledge in that case to solve the problem.
- *Revise* the proposed solution if necessary.
- *Retain* the parts of this experience likely to be useful for future problem solving.

These steps are illustrated in Figure 2.

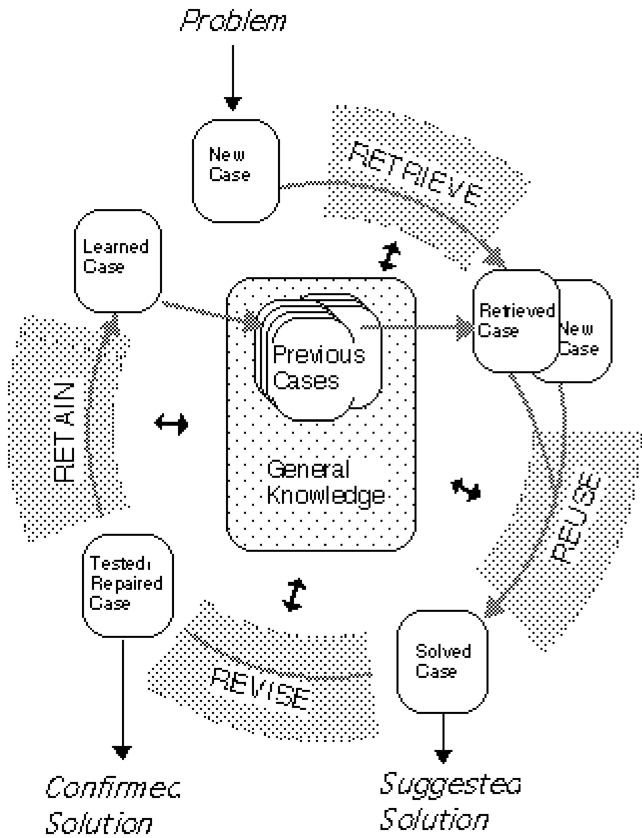


Figure 2. CBR cycle. (Figure is copied from (Aamodt and Plaza 1994) with kind permission of Agnar Aamodt. Downloaded on November 2, 1999 from http://www.iiia.csic.es/People/enric/AICom.html#RTFToC11)

1.3.1 Approaches to case-based reasoning

There are two basic approaches to CBR: a cognitive science based approach and a technology based approach. In the cognitive science based approach towards CBR, the goal is to explain how intelligence works. This view is expressed in the following statement.

> Real thinking has nothing to do with logic at all. Real thinking means retrieval of the right information at the right time. (Riesbeck and Schank 1989).

Cognitive scientists use CBR in an effort to deconstruct thinking. To the degree that CBR imitates thought processes, CBR models thought.

Kolodner (1993) surveyed 82 CBR systems. [3] Kolodner defines a case as

> a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner.

Kolodner's definition of a case is applicable in this thesis. Kolodner describes case-based reasoning as

> both a cognitively plausible model of reasoning and a method for building intelligent systems.

Leake (1996) identifies four elements of CBR:

> Case-based reasoning = retrieval + analogy + adaptation + learning

According to Leake, "CBR is fundamentally analogical reasoning." Leake explains that the difference between CBR and analogy is mostly a matter of approach.

> Research on analogy was originally concerned with abstract knowledge and structural similarity, while research on CBR is more concerned with forming correspondences between specific episodes based on pragmatic considerations about the usefulness of the result.

Leake (1996) identifies five main problems in AI that can be improved by CBR: knowledge acquisition, knowledge maintenance, increasing problem-solving efficiency, increasing quality of solutions, and user acceptance. Leake explains how CBR attempts to avoid such knowledge-related problems by assuming that there are few domain rules.

> Reasoning is often modeled as a process that draws conclusions by chaining together generalized rules, starting from scratch. CBR takes a very different view. In CBR, the primary knowledge source is not generalized rules but a memory of stored cases recording specific prior episodes. In CBR, new solutions are generated not by chaining, but by retrieving the most relevant cases from memory and adapting them to fit the new situations. Thus in CBR, reasoning is based on remembering … remindings facilitate human reasoning in many contexts and for many tasks, ranging from children's simple reasoning to expert decision-making.

Leake explains that CBR is based on the tenet that

> the world is regular: similar problems have similar solutions. Consequently, solutions for similar prior problems are a useful starting point for new problem-solving.

---

[3] All of the CBR systems surveyed by Kolodner (1993) come from research based in the United States, a fact subsequently pointed out by European researchers (López de Mántaras and Plaza 1997). There was apparently in the early 1990's a little rivalry between the "first school" of cognitive science based CBR researchers, based in the United States, and the "second school" of application oriented CBR researchers, based in Europe.

For anyone building a CBR application, this begs the questions: Who's remindings are most valuable? How is the world regular? Who is most qualified to discern similarity?… Presumably, knowledgeable people, or experts.

In the technology-oriented approach towards CBR, the goal is to construct useful decision support systems, as opposed to deconstructing thought. Technology is applied science, not pure science.

Technologists build systems from whatever is useful. Problem-specific knowledge is useful for building decision support systems. Therefore, technologists use knowledge acquisition strategies to build CBR systems. Over the past ten years, the technology-oriented approach has gained momentum. The recent "trend emphasizes the increasing importance of issues and techniques in the development of knowledge intensive CBR systems." (Aamodt and Plaza 1994).

CBR was originally proposed as an AI method to avoid the knowledge acquisition problem, the bottleneck in expert system development. CBR has been quite successful, as attested to by the reviews of Riesbeck and Schank (1989), Kolodner (1993), Leake (1996), and López de Mántaras and Plaza (1997). However, it has become increasingly clear in the literature that domain knowledge is valuable for technology-oriented CBR. As Aha (1998) puts it, "Knowledge engineering has been recast as case engineering."

1.3.2 Challenges for case-based reasoning

Knowledge acquisition is a fundamental challenge for CBR system developers. Developers must acquire knowledge about how to index and match cases, how to adapt cases into solutions, and how to repair failed solutions. Such knowledge enables us to build the "CBR knowledge base" shown in Figure 1.

Leake (1996) identifies four challenges for CBR research:

- Case adaptation: developing methods to convert imperfectly analogous cases into useful solutions.
- Case authoring: developing methods for preparing cases for inclusion in a case base, e.g., developing tools to enable an expert to participate directly in the case acquisition and case engineering process.
- Scaling up systems to large problems.
- Problems with libraries of many cases.

Improving the processes of case adaptation and case authoring are probably the most significant challenges in CBR today. Both processes depend on knowledge acquisition. Leake (1996) describes the CBR challenge of case adaptation as follows.

> Central questions for adaptation are which aspects of a situation to adapt, which changes are reasonable for adapting them, and how to control the adaptation process. Answering these questions may require considerable domain knowledge, which in turn raises the questions of how to acquire that knowledge. Many CBR systems depend on that knowledge being encoded *a priori* into rule-based production systems. Unfortunately, this approach raises the same types of knowledge acquisition issues that CBR was aimed at avoiding. It has proven a serious impediment to automatic adaptation.

Leake describes various methods for improving adaptation that divide into roughly two types: direct and indirect. Direct methods focus on the knowledge or methods used during adaptation. Indirect methods decrease the need for adaptation by retrieving cases that require less adaptation. If neither of these methods can be made to work, then the CBR system will enter into an endless loop and fail (see loop in Figure 1).

Adaptation is a main challenge for CBR. Indirect methods for avoiding adaptation decrease the need for adaptation by retrieving cases that require less adaptation (Leake 1996). Expertise about degree of feature salience can help us to avoid the need for adaptation. Riesbeck (1996) emphasizes that what sets CBR apart from rule-based reasoning is the presence of two processes—partial matching and adaptation— and describes the mixed status of adaptation in CBR as follows.

> On the one hand, adaptation is the ‘reasoning’ part of ‘case-based reasoning.’ Furthermore, most early CBR work focussed on the development and application of adaptation strategies, such as parameterization and abstraction/respecialization (Riesbeck and Schank 1989). On the other hand, adaptation is usually the weak link in a CBR system. Adaptation techniques are hard to generalize, hard to implement, and quick to break. Furthermore adaptation is often unnecessary. The originally retrieved case is often as useful to a human as any half-baked adaptation of it.

The fuzzy *k*-nn algorithm performs effective partial matching with a large database, composes solutions based on a weighted median of cases (cases weighted according to their degree of similarity) and, thereby, reduces the need for adaptation. We lessen the need for adaptation by scaling up to a large database of raw cases using a suitably designed fuzzy *k*-nn similarity measuring algorithm (Hansen and Riordan 1998).Case authoring is the process of preparing cases for inclusion in a case base. Aha (1997) describes the CBR challenge of case authoring as follows.

> CBR is not a magic bullet for the expert systems community. It is a technology that demands attention to the process of *case engineering*, which bears resemblance to

> knowledge engineering, [and because of inherent problems in case engineering] simplifying the case authoring task is of great practical value to prospective clients of commercial CBR tools.

In switching from knowledge engineering to case engineering, developers trade the problem of handcrafting rules for the problem of handcrafting cases. A main problem in implementing CBR is to build the *Case Base*, as shown back in Figure 1.

The requirement for application-specific knowledge to handcraft cases creates a bottleneck in CBR development. Domain experts are prohibitively expensive to employ for the construction and maintenance of decision support systems.

1.3.3 Retrieval of similar cases

Retrieval is the first and most important process in case-based reasoning. Case-based reasoning begins with cases and cases are obtained by retrieval. The basic problem in retrieval is to find similar cases—*to find good analogs.* The same problem challenges meteorologists who try to apply the technique of “analog forecasting” for the problem of weather prediction. We focus on how to improve the process of finding good analogs because the obtainment of good analogs will reduce the need for adaptation and the dependency on case authoring.

Improving retrieval is an open problem in CBR research and CBR system development (Leake 1996). How do we select past cases that best match the present problem? Such selection depends on being able to identify and evaluate relevant attributes and being able to perform partial matching between cases. Improving adaptation is another open problem in CBR. How do we adapt past cases that either do not agree perfectly with the current problem or do not agree with each other? Such adaptation depends on being able to make the best possible use of imperfect analogs.

In this thesis, we propose using a variation of the fuzzy *k*-nearest neighbors (fuzzy *k*-nn) method described by Keller et al. (1985) to enable a reasoner to identify and evaluate relevant features based on the experience of a domain expert. Experts evaluate and describe similarity fluently using a fuzzy vocabulary. For example, they might say, “Two attributes are *slightly* similar if the difference between their values is *near* 10.” Eliciting such knowledge from experts and encoding it in fuzzy sets enables the fuzzy *k*-nn method to emulate a discriminating expert at the task of finding similar cases.

Aha (1998) explains that feature weighting is the main challenge in developing *k*-nn algorithms. [4] He suggests that domain knowledge can assist *k*-nn algorithm development to weight features and to select relevant features or combinations of features. This thesis explains how a fuzzy *k*-nn technique helps us to obtain and use such knowledge about feature salience for determination of similarity. According to Luger and Stubblefield (1998):

> One of the most subtle and critical issues raised by CBR is the question of defining similarity. Although the notion that similarity is a function of the number of features that two cases have in common is quite reasonable, it masks a number of profound subtleties. For example, most objects and situations have an infinite number of potential descriptive properties; case-based reasoners typically select cases on the basis of a tiny retrieval vocabulary. Typically, case-based reasoners require that the knowledge engineer define an appropriate vocabulary of highly relevant features. Although there has been work on enabling a reasoner to determine relevant features from its own experience, determining relevance remains a difficult problem.

The fuzzy *k*-nn system queries a database using whichever potential descriptive properties best fit the present situation. The fuzzy *k*-nn system, rather than learning about important similarities and determining relevance "from its own experience," is taught opportunistically by a domain expert who is already well experienced at comparing attributes of cases and able to fluently describe important similarities with fuzzy words.

1.4 Fuzzy logic

In the previous section, we explained how CBR depends on retrieval of similar cases. In this section, we give a general introduction to fuzzy logic and explain how it can be used to achieve retrieval of similar cases. We do not deal with fuzzy logic in depth—many books and articles have done this already. All the fuzzy logic methods used in this thesis are explained in detail by Zimmerman (1991).

---

[4] Aha (1998) explains that *k*-nearest neighbor (*k*-nn) classifiers that use similarity functions to answer queries are a frequently studied group of "lazy learners." Aha (1998) divides learning algorithms into two categories: *eager* and *lazy*. Eager learning algorithms process data before receiving queries and any related new data. Such processing converts large amounts of data into compact abstractions such as rule sets, decision trees, or neural networks. Whereas, lazy learning algorithms process data after receiving queries and any new data, and can thus take advantage of last-minute information.

The fuzzy *k*-nn system described in this thesis may be viewed as a pure lazy learner. The system uses precise information about present cases—information that is only available in the context of the present—to inform the search for similar past cases. Anyone who is asked to find similar cases would naturally ask: Similar to what? The present case answers that question and is the basis of the query.

Fuzzy logic is an established methodology that is widely used to model systems in which variables are continuous, imprecise, or ambiguous. [5] The main idea of fuzzy logic is that items in the real world are better described by having partial membership in complementary sets than by having complete membership in exclusive sets. [6] This has the effect of increasing the resolution and the fidelity of categorization.

For example, suppose we can assign people into two sets, *short* and *tall*. In classical logic (i.e., non-fuzzy or "crisp" logic) an arbitrary threshold is specified. For instance, someone who is shorter than 160 cm is deemed to be *short*, and someone who is 160 cm or taller is deemed to be *tall*. Using this logic, one would conclude that two people who are of nearly identical height, 159 cm and 160 cm, fall into opposite categories of height, one *short* and the other *tall*. This is not how people think.

Whereas, in using fuzzy logic, an item may have partial membership in two or more sets. Someone who is 160 cm can have 0.5 degree membership in the *short* set and 0.5 degree membership in the *tall* set. For different heights, memberships can range continuously from 0.0 to 1.0 to accord with human perception. Fuzzy logic models how people think.

A fuzzy logic based methodology is used in this thesis for the following reasons.

- Fuzzy logic is effective for *eliciting and encoding knowledge* from domain experts (Kantrowitz et al. 1997). For instance, such knowledge can control recognition of similarity between two weather situations (Hansen 1997).
- Fuzzy logic is well-suited to *modelling continuous, real-world systems*. Many systems dealing with environmental data use fuzzy logic (Hansen et al. 1999).
- Fuzzy logic has a "*tolerance for imprecision* which can be exploited to achieve tractability, robustness, low solution cost, and better rapport with reality." (Zadeh 1999).

Zadeh (1999) explains the third point in an article entitled *From computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions*. Zadeh's vision is an inspiration for the fuzzy *k*-nearest neighbors technique described in this thesis. Zadeh (1996) expresses optimism for fuzzy logic partnering with other techniques, such

---

[5] Fuzzy logic is used in thousands of applications, in areas such as: transportation, automobiles, consumer electronics, robotics, computers, computers, telecommunications, agriculture, medicine, management, and education (Munakata and Jani 1994).

[6] Zadeh (1965) first defined a fuzzy set as follows: "A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function which assigns to each object a membership ranging between zero and one."

as machine learning theory and chaotic systems analysis, both of which are touched on in this thesis.

Fuzzy logic is especially useful for CBR because: CBR is fundamentally analogical reasoning (Leake 1996), analogical reasoning can operate with linguistic expressions, and fuzzy logic is designed to operate with linguistic expressions.

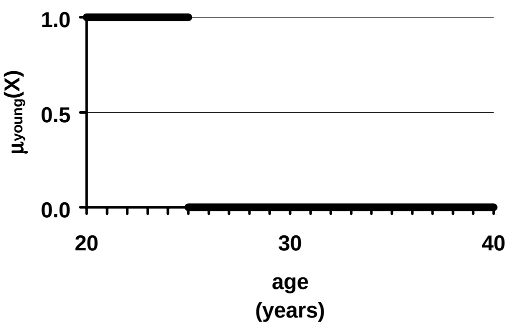
Fuzzy logic operates with linguistic, realistic variables, whereas classical logic operates with Boolean, discrete variables. A database query example illustrates the difference between the two. Suppose a marketing business is interested in identifying employees who have high potential. It could search its employee database for all employees who are *young* and who have *high sales*. Two approaches to querying the database are crisp range based and fuzzy set based.

The crisp approach is depicted in Figure 3 (a) (b) and (c). With the crisp approach, one specifies a discrete range based query as follows:

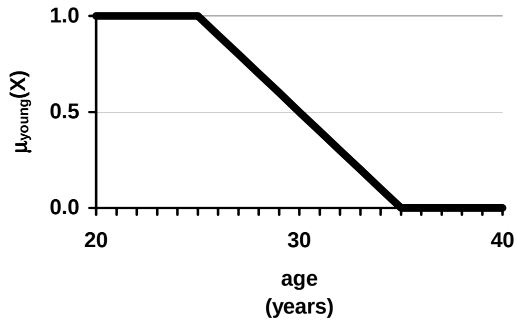
$$\text{young} \quad \Leftrightarrow \quad \text{age} \leq 25 \text{ years}$$
$$\text{high sales} \quad \Leftrightarrow \quad \text{sales} \geq \$500{,}000 \text{ per year}$$

If there is an employee who is 26 years old who averages $1 million per year in sales, the crisp search would fail to identify this employee. This employee has “zero membership” in the overlap of the specified crisp sets. Yet, most people would reasonably think that this person is young and has high sales.

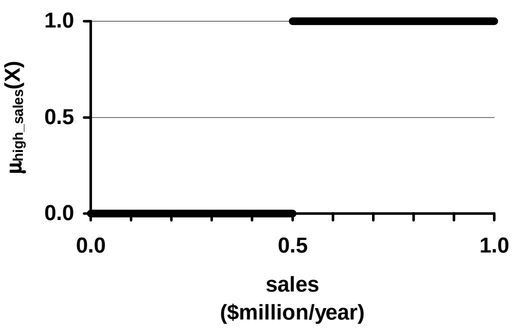
The fuzzy approach is depicted in Figure 3 (d) (e) and (f). With the fuzzy approach, one specifies a fuzzy set based query, in which fuzzy sets determine degree of membership in the sets *young* and *high sales*.



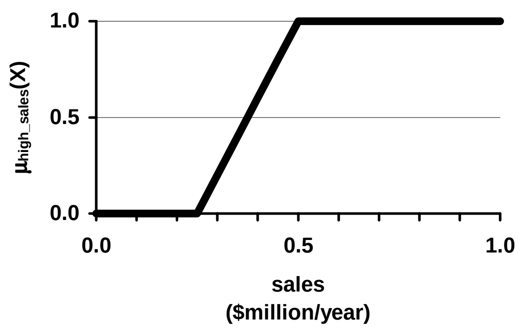
(a) Crisp set for young.



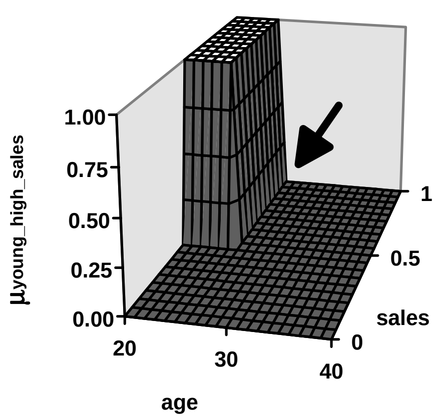
(d) Fuzzy set for *young*.



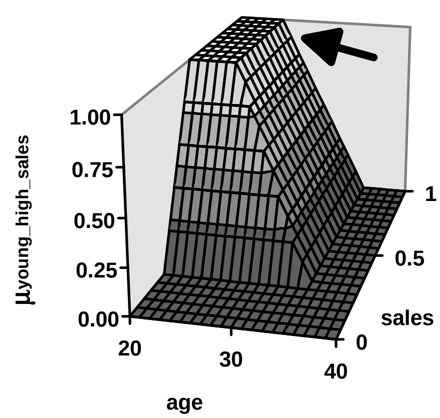
(b) Crisp set for *high sales*.



(e) Fuzzy set for *high sales*.



(c) Crisp decision surface for *young* with *high sales*.



(f) Fuzzy decision surface for *young* with *high sales*.

Figure 3. Crisp sets and fuzzy sets. Functions for dual membership in two sets. Arrows show how a 26-year old million-dollar-selling employee is accorded different levels of membership in the set *young with high sales* by crisp sets and by fuzzy sets. Using crisp sets, membership equals zero, whereas using fuzzy sets membership equals 0.9. The latter membership is more consistent with how people think.

The dual membership in the sets of those *young* with *high sales* for the 26-year-old, million-dollar-selling employee is calculated as follows.

$$max\{\mu_{young}, \mu_{high\ sales}\} = max\{0.9, 1.0\} = 0.9$$

The employee has 0.9 degree of membership in the specified fuzzy sets. This is consistent with the view that most people would have that this person is young and has high sales. Knowledge of how to interpret and evaluate such attributes can be obtained by interviewing an expert and encoding their responses as fuzzy sets.

Srinivasan et al. (1994) describe four enhancements that fuzzy logic provides to a neural network, which could also be provided to CBR, as follows.

- "Amalgamation of different pieces of knowledge is possible by application of fuzzy rules."
- "A large scale-knowledge base can be effectively handled and reduced by fuzzy front-end processor, making [neural network] learning easy and fast—non-precise and context dependent knowledge is represented using fuzzy logic."
- "Recognition and learning from noisy data is possible."
- "The technique is robust in that only some rules in knowledge in fuzzy knowledge base require to be updated with changing input conditions, avoiding the need to retrain the neural network."

For instance, point number 4 implies, for CBR, that we could avoid the need to re-optimize the entire case-comparison weight vector every time a new predictive/selective piece of knowledge is added to a similarity measuring function.

Elicitation of domain knowledge is a basic and common application of fuzzy logic. The "Fuzzy Logic FAQ" explains how membership values can be determined through subjective evaluation and elicitation as follows.

> As fuzzy sets are usually intended to model people's cognitive states, they can be determined from either simple or sophisticated elicitation procedures. At the very least, subjects simply draw or otherwise specify different membership curves appropriate to a given problem. These subjects are typically experts in the problem area. Or they are given a more constrained set of possible curves from which they choose. Under more complex methods, users can be tested using psychological methods. (Downloaded on April 20, 2000 from http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq.html)

Fuzzy methods represent cases with any combination of words and numbers. The fuzzy *k*-nn technique retrieves similar cases by emulating a domain expert who understands and interprets similar cases. The main contribution of fuzzy logic to case-based reasoning (CBR) is that it enables us to use common words to directly acquire domain knowledge about feature salience. This knowledge enables us to retrieve a few most similar cases from a large database, which in turn helps us to avoid the problems of case adaptation and case authoring. When cases don't fit perfectly, as often they never will, a practical option may be to inspect many cases, select the few most similar cases, and make reasonable inferences.

1.4.1 Fuzzy logic enables retrieval of similar cases

When domain experts are presented with a set of three unique situations and asked to describe the similarity between the three pairs in the set, they are more likely to say "very, somewhat, and slightly" than to say "yes, no, and I don't know." The former response couches the description of similarity in uncertain words, or fuzzy words. The inherent uncertainty is due to fuzziness, not randomness.

Figure 4 shows how such words map to fuzzy sets and thereby enable fuzzy operations to emulate a domain expert in the task of comparison. For a simple example, consider the problem of describing the similarity of three weather situations where, for simplicity of illustration, each weather situation is described only by temperature. The fuzzy set shown in Figure 4 solves this problem.

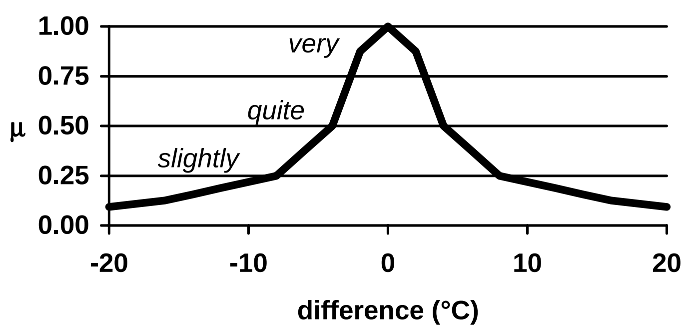


Figure 4. Fuzzy set to describe *degree of similarity* of temperatures as a function of the difference between the temperatures. Fuzzy set models cognitive state of expert weather forecaster who evaluates differences. Fuzzy set emulates expert at comparison.

The fuzzy set in Figure 4 is designed by interviewing an expert weather forecaster who is familiar with local effects. We ask the expert, "At what points do you consider two temperatures to be slightly similar, quite similar, and very similar?" The above fuzzy set maps a response of

"8°C, 4°C, and 2°C." The function is unimodal, continuous, and returns values in the range (0.0...1.0]. Fuzzy sets such as this are the basic component of the fuzzy *k*-nn technique.

Such fuzzy sets are a form of acquired knowledge. This is knowledge about relevance, feature salience, and importantly similar attributes. This is not knowledge in the form of rules, which is the commonest AI sense of knowledge.

Fuzzy sets allow a function, $\mu(x)$, to measure similarity between any attributes in the way that an expert would. For example:

humidity difference of 5%	$\rightarrow$	very similar	$\rightarrow$	$\mu(5) = 0.75$
humidity difference of 20%	$\rightarrow$	slightly similar	$\rightarrow$	$\mu(20) = 0.25$

In comparing cases composed of multiple attributes, attributes that are more important than others have narrower fuzzy sets. For example, wind direction affects local weather more strongly than temperature, so it should have a narrower, more discriminating fuzzy set.

A system equipped with such a similarity-measuring function can take the present temporal case and rate all the previous cases in terms of similarity. In practice, all cases have similarity scores: $0.0 < sim < 1.0$ This quality of the fuzzy *k*-nn technique reflects perception of real weather cases, which is that real weather cases are never identical and are never "totally dissimilar."

We combine fuzzy logic with CBR because fuzzy logic is helpful for acquiring knowledge and it provides methods for applying knowledge to real-world data. Fuzzy logic simplifies elicitation of knowledge from domain experts, such as knowledge of how similarity between two cases depends on the difference between their individual, collective, and temporal attributes. Fuzzy logic emulates human reasoning about similarity of real-world cases, which are fuzzy, that is, continuous and not discrete. For example, using fuzzy sets elicited from a weather forecaster who is experienced at comparing and evaluating similarity between weather cases, fuzzy logic emulates the forecaster at the task of recognizing good analogs.

1.5 Weather prediction

In the previous two sections, we introduced CBR and fuzzy logic and explained how these subjects relate to retrieval of similar cases. In this section, we briefly introduce weather prediction, describe a method of weather prediction called "analog forecasting" (a meteorological form of CBR), and explain how analog forecasting depends on retrieval of similar cases.

Fundamentally, there are only two methods to predict weather: the empirical approach and the dynamical approach (Lorenz 1969a). The empirical approach is based upon the occurrence of analogs (i.e., similar weather situations). The dynamical approach is based upon equations of the atmosphere and is commonly referred to as computer modeling. The empirical approach is useful for predicting local-scale weather if recorded cases are plentiful (e.g., cloud ceiling and visibility in a few square kilometres around an airport). Because of grid coarseness, the dynamical approach is only useful for modeling large-scale weather phenomena (e.g., general wind direction over a few thousand square kilometers).

Weather prediction is regarded by meteorologists as both a science and an art. Weather prediction relies upon objective techniques based upon decades of research, and it relies upon subjectivity and judgment based upon personal experience and local rules and practices. We will regard weather prediction as an objective process. Objective techniques are universal, whereas subjective techniques are local. Objective techniques are used consistently and are portable, whereas subjective techniques are used inconsistently: subjective techniques vary from person to person, from time to time, and from place to place. Analog forecasting is an objective method for weather prediction that makes predictions for a present weather situation based on the outcomes of similar past weather situations. Analog forecasting is the weather prediction technique that we aim to improve.

In subsection 1.5.1, we introduce the airport weather prediction problem addressed in this thesis. In subsection 1.5.2, describe the state of the art of artificial intelligence in weather prediction. In subsection 1.5.3, we describe the analog forecasting technique and explain how it depends on retrieval of similar cases.

1.5.1 Airport weather prediction problem

An airport weather prediction is a concise statement of the expected meteorological conditions at an airport during a specified period (US National Weather Service Aviation

Weather Center, 1999). An airport weather prediction is, in meteorology, commonly referred to as TAF, short for *Terminal Aerodrome Forecast*. When pilots give weather forecasts to passengers before landing, they are reading TAFs.

TAFs are made by expert forecasters. These experts have general knowledge about how large scale weather systems behave and specific knowledge about how local scale weather phenomena behave idiosyncratically at specific airports. Experts bridge the gap between simple persistence forecasting and NWP-assisted statistical forecasting on the local scale (Battan 1984).

The three types of forecasts most commonly made by forecasters are TAFs, public forecasts and marine forecasts. Of these, TAFs are the most precise and thus the most challenging type of forecast to make, both in terms of measurable weather conditions and in terms of timing. Forecasts of the height of low cloud ceiling are expected to be accurate to within 100 feet. Forecasts of the horizontal visibility on the ground, when there is dense obstruction to visibility, such as fog or snow, are expected to be accurate to within 400 metres. Forecasts of the time of change from one flying category to another are expected to be accurate to within one hour. In comparison, public and marine forecasts can be much less precise. For example, in public forecasts, it may be sufficient to predict "variable cloudiness this morning," and in marine forecasts, it may be sufficient to predict "fog patches forming this afternoon."

NAV CANADA [7] measures TAF performance in four ways, with three ceiling and visibility accuracy statistics [8] and with a speed-of-amendment statistic. The commonest cause for amendments is unforecast ceiling or visibility (Stanski 1999). So, accurate predictions of cloud ceiling and visibility are clearly important.

In this thesis, we are only directly concerned with the two qualities of TAFs that are routinely measured by NAV CANADA, which are as follows.

- *Accuracy of prediction of flying condition category*. Flying category determined by both cloud ceiling height and horizontal visibility, two obstructions to vision for pilots. The lower the forecast category, the more expensive precautions pilots must take.

---

[7] NAV CANADA is the agency that manages Canada's air navigation system—including air traffic control, flight information, weather briefings, airport advisory services and electronic aids to navigation.
[8] In the Experiments chapter, we will measure the accuracy of fuzzy *k*-nn based predictions with the same statistics that are used by NAV CANADA and described by Stanski et al. (1999).

- *Timeliness of revision.* This describes the length of time from the detection of weather conditions contradicting TAF (i.e., a forecast "going bust") and the delivery of a suitably revised forecast.

So, to improve the quality of airport weather predictions, the three main challenges are:

- Make airport weather forecasts more accurate.
- Make the forecasting process more efficient.
- Make analog forecasting more useful.

1.5.1.1 Motivations for improving airport weather prediction

The motivations for improving the airport weather prediction process are both ergonomic and economic. Airport weather forecasting is a difficult task for forecasters. A system that can provide forecasters with improved and timely guidance will help to make their work easier and thus help to make them more effective.

TAFs are economically important to TAF users, providers, and producers. Airlines use TAFs. In Canada, NAV CANADA provides TAFs to airlines and Environment Canada (EC) produces TAFs for NAV CANADA.

Accurate TAFs increase the safety of airplane passengers and the profitability of airlines. When "bad weather" [9] is forecast at the destination airport of an airplane, the pilot must load on extra fuel to ensure the airplane will be able to reach an "alternate airport" in case diversion en route becomes necessary. So, reliable forecasts of airport weather—"bad weather" and "good weather," at destinations and alternates—are important for the safety of airplane passengers.

At the same time, airlines do not want airplanes to carry more fuel than necessary for safety. It is expensive to fly fuel from one airport to another. Unused fuel on arrival is an unwanted expense. As TAF accuracy increases, the benefit to airlines increases. Leigh (1995) studied the of effect of TAF accuracy and concluded that "the economic benefit of a uniform, hypothetical increase in TAF accuracy of 1% is approximately $1.2 million [Australian] per year for Qantas international flights into Sydney."

---

[9] By "bad weather," we mean weather conditions that complicate flying, such as low cloud ceilings or low visibility. Low visibility is caused by various factors, such as fog or snow. Such conditions may slow or stop airplane traffic, just as fog affects the way motorists drive. Even though large airplanes can land with little visibility, such conditions commonly cause expensive traffic problems. Traffic problems at one airport can "ripple" through the entire air transportation network and cause scheduling problems for distant airports.

Patton (1996) interviewed airplane pilots to determine how they behave in response to government transportation regulations, airline policies, air traffic flow management, types of airplanes, and airport weather forecasts. Pilot behavior is complex to say the least [10], but it is clear from her investigation that inaccurate, pessimistic airport forecasts cause pilots to load on extra "unnecessary" fuel and that this directly increases operating costs for airlines.

Thirteen years ago, White (1987) reported that then-recent improvements in forecasts had enabled airlines served by the U.K. Meteorological Office to significantly reduce fuel consumption and thereby save an estimated £50 million per year.

There is a growing market for more accurate and more up-to-the-minute TAFs. White (1995), a director of the International Air Transport Association (IATA), identifies important economy-driven and computer-assisted trends in the aviation industry. The aviation industry contributes about $1 trillion [US] per year to the global economy and air travel is growing at a rate of about 6% per year. By equipping airplanes with the "latest space-age technology options to optimize performance," the IATA has the goal of establishing "an era of 'free flight,' meaning a kind of Utopian environment where aircraft can operate on a totally flexible 'flight plan' making optimum use of prevailing weather conditions and forecast updates. ATC [Air Traffic Control] will only intervene when necessary to prevent serious loss of separation."

In Canada alone, the production of TAF's accounts for about $5,000,000 a year revenue to Environment Canada (EC). [11] EC is contracted to provide accurate and timely predictions of ceiling and visibility to NAV CANADA. A system which could provide useful ceiling and visibility guidance to make TAF's, autonomously and using real-time data, would be helpful for EC. EC, like all public and private sector agencies, is under continuous pressure to economize (Doswell and Brooks 1998). TAFs are expensive to produce, presently costing about $30,000 per year per airport for round-the-clock coverage (Macdonald 1998).

TAFs are a good value for airlines. To estimate their value, Doyle (1995) developed a plausible scenario, incorporating several reasonable assumptions, in which forecast service for a

---

[10] Patton (1996) represents one common set of pilot decision-making behaviors with a decision tree with over 100 nodes.

[11] All financial figures in this thesis are in Canadian dollars unless stated otherwise. Environment Canada currently receives about $24 million in annual revenue from NAV CANADA for the provision of aviation weather services (Macdonald 1999). The provision of TAFs accounts for about 20% of this revenue (Meadows 1997). The contract between EC and NAV CANADA specifies accuracy and timeliness targets for TAFs. The current contract between EC and NAV CANADA will expire in November 2001, after which time NAV CANADA may seek tenders for aviation services. NAV CANADA is naturally concerned

set of airports is withdrawn, and calculated what the resultant extra costs would be to Air Canada. Extra costs would result because pilots, when filing flight plans, could not use the affected airports as "alternate" landing sites, and would therefore have to file more distant airports as alternate, and would therefore have to load on and fly more fuel, which is expensive to do. In the scenario, forecast service is withdrawn from nine airports and the estimated resultant extra costs to Air Canada are \$450,000 per year. The scenario does not take into account potential savings from not having to pay for TAF production costs of \$270,000 per year (= 9 $\times$ \$30,000). But even assuming TAF production costs could be subtracted from additional fuel-carrying costs, Air Canada would still lose \$270,000 per year in the scenario.

The scenario of (Doyle 1995) only accounted for additional fuel-carry costs to airlines resulting from removal of TAF coverage for alternate airports. There would certainly be two other additional costs:

- Diversions to remoter alternate airports would cause additional costs to airlines, such as lodging, transporting, and placating dissatisfied passengers.
- The loss of TAF coverage at any particular airport would increase planning difficulties for managers at that particular airport.

1.5.2 State of the art of AI in weather prediction

The state of the art of AI in weather prediction is advancing steadily. Recently, Christopherson (1998) surveyed the meteorology literature and identified over 40 AI-meteorology papers, whereas ten years ago a similar survey by Conway (1989) identified only 4 such papers.[12] Operational forecast systems using AI are now being used by the Meteorological Service of Canada (MSC), the U.S. Army, and the U.S. Navy (Christopherson 1998). In 1998, the American Meteorological Society gave its "stamp of approval" to AI by holding its First Conference on Artificial Intelligence.

Conway (1989) identified three special challenges which meteorology places on AI: [13]

---

with minimizing its costs and passing along any savings to its clients and stakeholders. It is logical to expect that NAV CANADA will want a better deal when they go to tender.

[12] The fact the Conway (1989) found many fewer AI-meteorology papers than Christopherson (1998) is partly explainable by the facts that: (1) Conway's survey focused on expert systems, rather than AI, and (2) some additional meteorology application papers identified by Conway, while "AI-like," were not billed as AI. Even so, it is obvious that AI-meteorology papers are becoming increasingly numerous.

[13] The first two challenges are not dealt with directly in this thesis. Convenience and speed are operational concerns. Pattern recognition problems relate mostly to image interpretation—Bezdek and Pal (1992) offer a large collection of papers which describe how to use fuzzy models for pattern recognition. We plan to attend to these special challenges in future related work.

- Need for convenience and speed.
- Pattern recognition problems.
- Missing and conflicting data.

Fuzzy techniques can assist in all of these challenges. Forecasters do not stop reasoning when they miss certain data, or have conflicting or ambiguous data; they continue to reason and attach an appropriate level of uncertainty to their conclusions. Conway (1989) explains how forecasters reason with inconclusive data as follows.

> As humans we do not normally reason in numerical terms but prefer vaguer notions of things being 'probable' or 'likely,' so the appropriate assignment of probabilities is one of the main difficulties of encoding human expertise in the form of rules. How best to deal with 'reasoning under uncertainty' is a subject of continuing research in the expert systems community.

Instead of trying to assign appropriate probabilities or to encode expertise in rules, which are difficult tasks, the fuzzy *k*-nn technique uses the following "vaguer notions."

- Similar weather situations (cases) evolve similarly.
- Similarity can be evaluated using fuzzy sets.

The driving force behind the development of AI-meteorology systems is the need to deal more effectively with the immense stream of data that forecasting depends upon. For example, we receive about 10 Megabytes per second of remotely sensed data from satellites. [14] NWP also produces huge amounts of data which needs to be incorporated together with other types of predictive information into forecasts. Weather forecasters need improved computer systems and AI systems to take better advantage of huge and ever-increasing amounts of data.

Klein (Dyer and Moninger 1988) identifies two problems facing developers of expert systems for weather prediction. First, the "genuine expert" may be difficult to find or identify (e.g., two alleged experts may contradict each other). Second, "there are pitfalls inherent in the practice of asking the expert to describe the unusual or difficult cases to the exclusion of ordinary events." Uncommon situations may be over-represented by the inference engine. This would emulate the occasional tendency of weather forecasters to "over-forecast weather." [15] Asking a

---

[14] Gershon and Miller (1993) estimated, "By the year 2000, satellites deployed by the National Aeronautics and Space Administration will be transmitting 1 terabyte of data to earth every day."

[15] Forecasters often apply complex forecasting techniques when simple short-tern persistence forecasting would probably produce more accurate forecasts. In a comparison of simple persistence forecasts with human-produced forecasts, Dallavalle and Dagostaro (1995) report that, "Generally persistence forecasts appeared to have higher skill than the local forecasts for the 3-hour projection." This tendency that people have to solve simple problems with all the tools at their disposal is referred to as "over-forecasting."

forecaster to describe all the difficult situations may lead to an unnecessarily complicated view of the forecasting process. CBR, or the fuzzy *k*-nn technique, can help us to avoid both problems, first, by reducing dependency on an expert and on knowledge acquisition, and, second, by giving all past cases an equal chance to affect the prediction for the present case.

Meyer (Frankel et al. 1995) suggests that "AI might have a role in assisting the forecaster in interpreting the output of numerical models [NWP] and adjusting it for local conditions." Mosher (1998) claims that, "Even with the new mesoscale forecast models, the meteorological forecaster can add value to the [NWP] guidance. The forecaster can provide unique information that is not available from [NWP]." Similarly, AI can combine unique data from complementary sources, such as airport weather archives and NWP. Fuzzy logic and the fuzzy *k*-nn technique are well-suited to combining and operating on heterogeneous types of data. [16]

Christopherson (1998) expresses optimism for the future of AI in meteorology when he concludes his survey as follows.

> The complexity of the modern weather forecasting (more datasets, modern information processing systems, larger areas of forecasting responsibility, shorter deadlines, more detailed forecasts), has many attributes which are defined for the application of AI. … Forecasters need assistance to more fully utilize this "data flood" and develop the modern forecasting process. AI techniques, particularly expert systems and neural networks, offer solutions to these problems.

Christopherson (1998) qualifies his optimism by describing hurdles for AI acceptance and use in weather forecasting as follows.

1.5.2.1 Why meteorologists have rarely used artificial intelligence

Based on extensive consultation with forecaster-developers and meteorological researchers working in AI development, and on a survey of over 40 AI-meteorology systems, Christopherson (1998) lists probable causes for the limited acceptance and use of AI in operational forecast offices as follows.

1. "The lack of specific, national level plans to integrate such technologies into the forecast process."
2. "The lack of a single computer environment *in the field* that has the power and flexible access to integrate diverse, complex or very large, non-static datasets. This is especially true because

---

[16] Fuzzy rules typically operate on variables of different dimensions. For example, a rule for a furnace could be: *if temperature is low and pressure is low then increase heat*. Temperature and pressure are expressed with different physical units, but the fuzzy rule operates on equally communicative fuzzy set representations of the variables.

domain experts (meteorologists) are rarely system programmers, or academically trained in AI."

3. "The AI development process is inherently an *engineering process* rather than a scientific investigation. Meteorologists are trained to do the latter and commonly avoid or even scorn the former."
4. "AI is often non-linear (subtle changes in input yield large changes in output). It is also not based on a physical model of the problem domain. This deters meteorologists, who want *algorithmic* solutions that model the atmosphere."
5. "While AI techniques are a broad and versatile technology, most applications solve narrow problems. Changing anything requires an entire developmental re-work to regain any skill and may require substantial changes in system design."
6. "The often specific nature of AI solutions suggest they are best used at the [regional] forecast office rather than the national center. However, the forecast office has traditionally been ill equipped to work with sufficiently detailed data sets and there has been a lack of sufficiently detailed data sets."
7. "AI will not be accepted until it is developed, taught, and used in university and college meteorology programs and government research laboratories and training facilities."

Considering points number 3 and 4 together helps to explain why, in meteorological research, interest in the development of the analog forecasting technique has been almost completely displaced by interest in computer modeling (NWP). Over the past 30 years, NWP has come to dominate many meteorological research agendas. [17] At the same time, little innovation has been attempted with analog forecasting techniques. All of the references to analog forecasting in the meteorological literature of the past 30 years rely heavily on statistical techniques, techniques that have hardly changed over the past 30 years. [18]

1.5.2.2 Why meteorologists need decision support systems

By interviewing forecasters, Kumar et al. (1994) found four reasons why forecasters need decision support systems, which are paraphrased as follows.

- Forecasters are challenged in their present work setting to absorb, comprehend, and remember a large amount of information which arrives in a continuous stream. Tight

[17] Dyer and Moninger (1988), from a workshop on AI research in environmental sciences, note that "One speaker said that weather forecasting R&D in the 1940's and 1950's shared much in common with the current AI thrusts, but that research was essentially shelved for 30 years as numerical weather prediction (NWP) took prominence.". Frankel et al. (1993), from another workshop on AI needs in meteorology, note that a representative of the Meteorological Service of Canada emphasized the prominence of NWP: "A major thrust in Canadian meteorological research is the continued development of world class NWP models."

[18] The section *Additional references on analog forecasting in meteorology* lists meteorology papers which describe the challenges of implementing analog forecasting.

deadlines exacerbate the problem. As a result, forecasters sometimes make "errors in judgment."

- It is difficult to discover through forecasting experience how to make near-optimal forecasts.
- Forecasters themselves express uncertainty about how to best use available forecast guidance information. Even experienced forecasters do not know how to best use guidance information.
- Some forecast verification statistics do not show any improvement in forecast skills over recent decades despite improvements in the quality and quantity of guidance information over the same time interval.

Kumar et al. (1994) used the machine learning technique of inductive learning to obtain prediction rules. These prediction rules were the basis of a system to predict 24-hour rainfall in Melbourne City, Australia. Their problem was to make *categorical* predictions of rainfall during a 24-hour period in Melbourne City Australia. They had a 30-year set. They used up to 129 attributes. Of these attributes, 59 were from NWP prognostic fields. So, they combined climatological and NWP guidance. They used inductive learning programs to build decision trees. [19] The output of the learning programs was represented as sets of rules and forecasters were asked to comment on these rules.

> According to the forecasters, even though the induction methods performed slightly better than the current prediction method, it is much easier for the forecasters to understand and use the automatically generated symbolic production rules by the induction method than the current complex statistical method, to perform the forecasting operations. [20]

Compared to statistical methods, machine learning has a good *explanation capability*, a desired quality in AI systems. It promotes user acceptance. Given a choice, users seem to prefer "transparent systems" over "black boxes"—scrutability over inscrutability. The fuzzy *k*-nn method should appeal to users in the same way. Its solutions are composed from actual cases—cases which can be presented to users to scrutinize if they so desire.

---

[19] Kumar et al. (1994) used the commercially available packages C4.5 and ID3. The system-specific details of how their systems built decision trees are not relevant in this thesis. What is relevant is that they used decision trees that split data into crisp sets. Our fuzzy k-nn algorithm does not split data into crisp sets and, therefore, appears to be unique among weather forecasting applications.

[20] Apparently, from (Kumar et al. 1994), the current prediction method for Melbourne is a sort of "man-machine mix." Forecasters appraise, select, and use whatever information or guidance is available, and the most relied-upon guidance is statistical (climatology plus NWP).

1.5.3 Analog forecasting: An empirical weather prediction technique that depends on retrieval of similar cases

Weather patterns repeat themselves—this is the basic idea behind the weather prediction technique called *analog forecasting*. Analog forecasting is a meteorological form of CBR. Analog forecasting is simple in theory: *make a prediction for the current situation based on the outcome of similar past situations*. However, development of analog forecasting systems is challenging in practice.

Analog forecasting is by far the oldest weather prediction technique. Useful weather sayings are based on recurring patterns of weather, and using recurring patterns of weather is essentially analog forecasting, thus useful weather sayings are a form of analog forecasting. For example, the following familiar saying is at least 2000 years old. [21]

> Red sky in the morning, sailors take warning.
> Red sky at night, sailors delight. (Anonymous)

The *Online Guide to Weather Forecasting* [22] describes analog forecasting as follows.

> It involves examining today's forecast scenario and remembering a day in the past when the weather scenario looked very similar (an analog). The forecaster would predict that the weather in this forecast will behave the same as it did in the past. … The analog method is difficult to use because it is virtually impossible to find a perfect analog. Various weather features rarely align themselves in the same locations they were in the previous time. Even small differences between the current time and the analog can lead to very different results. However, as time passes and more weather data is archived, the chances of finding a 'good match' analog for the current weather situation should improve, and so should analog forecasts.

In a practical sense, the fuzzy *k*-nn method learns as cases accumulate. As databases of cases increase in size, the chance of finding good analogs for any given weather situation increases.

To use analog forecasting, we must find good analogs and we must use these analogs appropriately. The two main challenges are:

- Develop a good similarity metric.
- Determine confidence intervals and practical time scales for analog predictions.

---

[21] "When it is evening, ye say, 'It will be fair weather: for the sky is red.' And in the morning, 'It will be stormy today, for the sky is red and lowering.'" (Matthew 16:2-3)
[22] The above description of analog forecasting is from "Online Guide to Weather Forecasting" at the University of Illinois Department of Atmospheric Sciences (http://ww2010.atmos.uiuc.edu/(Gl)/guides/mtr/fcst/mth/oth.rxml, downloaded on March 25, 1999).

1.5.3.1 Investigations into feasibility of analog forecasting uncovered chaos

The work of Edward Lorenz is seminal in the modern “science of chaos.” In the past, *chaotic* commonly meant disorder, uncertainty or randomness. Increasingly, *chaotic* describes a special type of order observed in systems in real-world areas such as physics, economics, statistics, chemistry, engineering, biology, and medicine (Abarbanel et al. 1993).

Lorenz (1963, 1969b, 1977, and 1993) tested the feasibility of using models, based on either analog forecasting or on physical equations of the atmosphere, to produce long-range weather forecasts. Lorenz reasoned that because weather obeys deterministic physical laws, if one could initialize a weather model perfectly, then deterministic long-range weather forecasting would be possible. He noticed that tiny errors in model initializations grow exponentially into large errors as models run forward in time. In the real world, such errors are inevitable because of practical limitations on weather measurement precision and accuracy. Model initialization errors limit the time range of both analog forecasting systems and numerical weather prediction systems. Lorenz (1963) concluded that the results of his experiments indicated that:

> prediction of the sufficiently distant future is impossible by any method, unless the present conditions are known exactly. In view of the inevitable inaccuracy and incompleteness of weather observations, precise very-long-range forecasting would seem to be non-existent.

Lorenz explained what his results implied for the feasibility of long-range analog forecasting: long-range, global weather prediction is infeasible because the atmosphere is sensitively dependent on initial conditions. [23] For deterministic weather prediction to be possible, one would have to find a *perfect* analog for the present case. However, because the global atmosphere exists in innumerable states, there are two impediments for analog forecasting. First, it is highly unlikely that perfect analogs ever exist. [24] Second, even if perfect analogs did exist, it is technically impossible to measure the atmosphere to the required level of precision. Lorenz (1963) noted that “these conclusions do not depend upon whether or not the atmosphere is deterministic.” Today, Lorenz’ conclusions are accepted in meteorology.

---

[23] In Lorenz' experiments, “long-term” pertained to prediction of upper atmospheric fields days in advance. Aviation weather changes more rapidly, so in the context of this thesis, predictions six to twelve hours in advance can be thought of as “long-term.”

[24] Van den Dool (1994) estimated that it would take on the order of “10^{30} years to find analogues that match over the entire Northern Hemisphere 500 mb height field to within current observational error.” The 500 mb (millibar) height field is a pressure field in the upper atmosphere, at about 6 km altitude, which is commonly examined by weather forecasters.

Reflecting on over three decades of research, Lorenz (1993) concludes, “We are left with the strong impression that the atmosphere is chaotic, but we would like additional evidence.” We assume that weather is chaotic and that it behaves accordingly, that is, that the flow of weather is sensitively dependent on initial conditions.

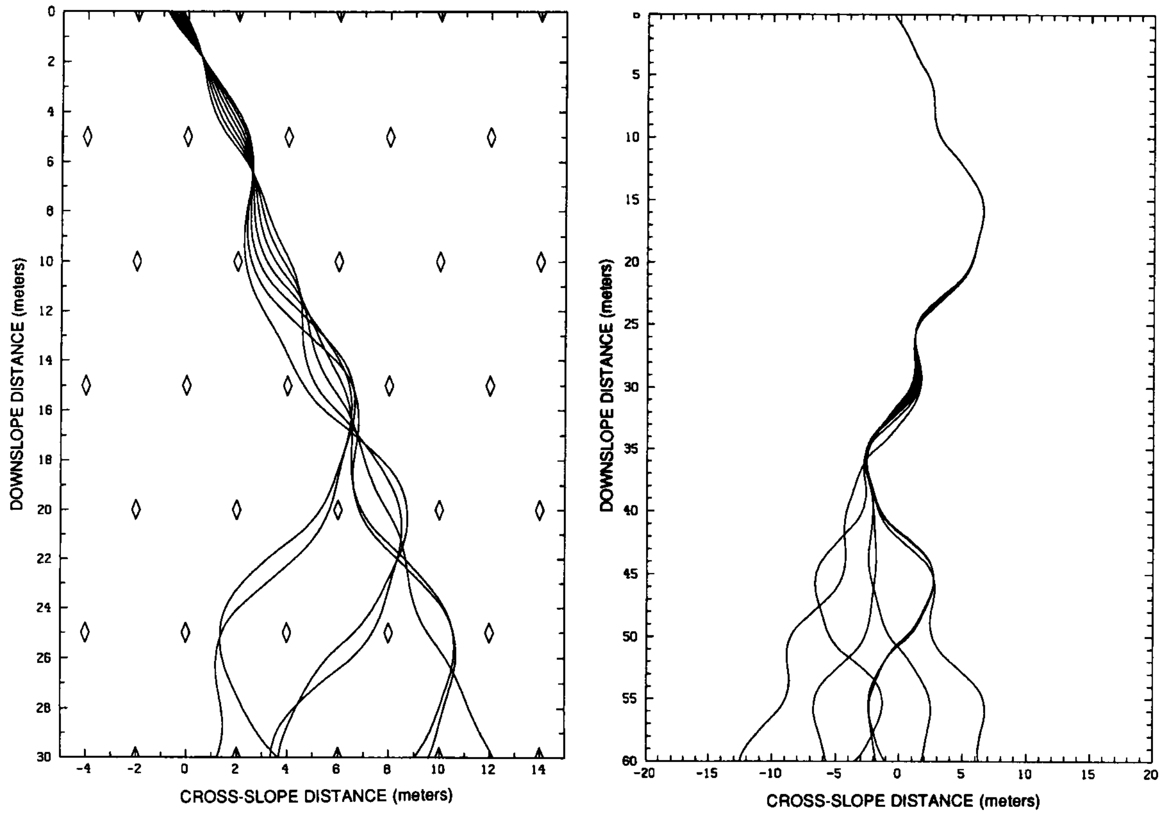
1.5.3.2 Temporal cases are chaotic trajectories

A temporal case is a short segment of a long record of a multidimensional, real-world process. In this thesis, and in general, a temporal case can describe any recorded, real-world process. In theory, if two distinct temporal cases are identical, then the sequences of events following those two cases will be identical. If one of those cases describes the present situation, then the problem of prediction is deterministic—one simply predicts a recurrence of the sequence of events that followed the previous identical case. However, in reality, identical cases are usually rare and case-based prediction is seldom so simple. There are fundamental practical limitations on case-based prediction method. These limitations are described in this section. These limitations will be reiterated in subsection 2.2.1 (pg. 48) when we review a fuzzy logic based formalism for deterministic CBR, proposed by Dubois et al. (1997), that is based on the principle: *“The more similar are the problem description attributes, the more similar are the outcome attributes.”*

In weather prediction, the method closest to CBR is analog forecasting. Analog forecasting is based on the principle that the more similar the current weather situation is to a past weather situation, the more similar the upcoming weather will be to that which followed the past weather situation. In its strongest form, this principle implies deterministic weather prediction.

In the real-world, chaos prevents determinism. Chaos imposes fundamental limitations on the applicability of case-based reasoning for predicting physical processes. Small differences between the initial states of two systems tend to grow exponentially over time and the two systems become increasingly dissimilar.

A good way to appreciate chaos, and the implications of a system being sensitively dependent on initial conditions, is with the following demonstration by Lorenz (1993). Suppose a snowboard starts sliding down a bumpy ski slope from a certain position at a certain velocity. The snowboard slides freely down the slope, curving left and right as it swerves away from bumps on the slope. A number of such trajectories are illustrated in Figure 5. Each such trajectory is a specific temporal case.



(a) Initial displacements spaced at 10 cm intervals.

(b) Initial displacements spaced at 1 mm intervals.

Figure 5. Chaotic: sensitively dependent on initial conditions. Each figure shows the paths of seven snowboards crossing a starting line with varying initial displacements and identical velocities. Diamonds in Figure 5 (a) indicate centers of bumps in snow. (Figures are copied from (Lorenz 1993) with kind permission of the University of Washington Press.)

The effects of varying initial conditions on the trajectory was tested by Lorenz with computer simulations. Each snowboard began on the starting line with the same velocity, both forwards and sideways components. The only condition made to vary was *x*, the position of the snowboard on the starting line. In the first set of trials, initial displacements were incremented by 10 cm, as shown in Figure 5 (a). In the second set of trials, initial displacements were incremented by only 1 mm, as shown in Figure 5 (b).

The results clearly show how the trajectory is *sensitively dependent on initial conditions*. The more similar the initial conditions of two temporal cases are, the more similar the outcomes are. Given the initial coordinates of a ski on the slope and a case base of past complete trajectories, it is possible to predict the path of the ski using an analog forecasting method: Make predictions based on the outcomes of similar past situations.

The snowboard analogy shows how analog forecasting works and how it fails. Differences between the states of two systems, which are initially similar, tend to grow exponentially over time. This implies that there is a practical limit on the time range of analog forecasting. Comparing Figure 5 (a) with Figure 5 (b), note that even though the initial conditions in (b) are 100 times closer than in (a), the tracks in (b) remain similar for only about twice the distance as in (a). The tracks in (a) diverge sharply after about 20 metres and the tracks in (b) diverge sharply after about 40 metres. Small initial differences tend to grow exponentially over time. So, in a chaotic environment, any attempt to improve analog forecasting by using increasingly similar cases will yield diminishing returns.

Confidence in the predictions depends on the distribution of the analogs. So long as the analogs are packed together, the predictions are precise and reliable. After the analogs fan apart, the predictions are vague and unreliable.

To use an analog forecasting method appropriately, the main limitation to recognize is the *practical time range.* We must identify the point in time when analogs become so dissimilar from each other that analog forecasts become unreliable. In meteorology, this concept is referred to as a “limit of predictability.”

1.5.3.3 State of the art of chaotic data analysis

This subsection simply highlights some observations made by Abarbanel et al. (1993) in a review entitled *The analysis of observed chaotic data in physical systems*. These observations are both motivational and instructive for this thesis.

- “In a sense we shall describe new methods for the analysis of time series, but on another level, we shall be providing handles for the investigation and exploitation of aspects of physical processes that could simply be dismissed as ‘stochastic’ or random when seen with different tools. Indeed the view we take in this review is that chaos is not an aspect of physical systems to be located and discarded, but is an attribute of physical behavior that is quite common and whose utilization for science and technology is just beginning. The tools we discuss here are likely also to be just the beginning of what we can hope to bring to bear in the understating and use of this remarkable feature of physical dynamics.”
- “This article is designed to bring scientists and engineers a familiarity with developments in the area of model building based on signals from nonlinear systems. The key fact that

makes this pursuit qualitatively different from conventional time series analysis is that, because of the nonlinearity of the systems involved, the familiar and critical tool of Fourier analysis does very little, if any, good in the subject. The Fourier transform of a linear system changes what might be a tedious set of differential equations into an algebraic problem where decades of matrix analysis can be brought to bear. Fourier analysis of nonlinear systems turns differential equations in time into integrals in frequency space involving convolutions among the Fourier transforms of the dependent variables. This is rarely an improvement, so Fourier models are to be discounted at the outset, though as an initial window through which to view the data, they may prove useful."

- "It is not uncommon to see attempts to overcome the limitations imposed by small data sets by measuring the system more frequently. … this is not an effective tactic. The raw number of points is not what matters; it is the number of trajectory segments, how many different times any particular locale of state space is visited by an evolving trajectory, that counts."
- "Our task is to find points in our sample library that are very close together and watch how trajectories specified by following points separate. In locating the initial neighboring points we must not consider points that are from the same temporal segment of the library."
- [Regarding local modelling] "We now assume that our data are embedded in an appropriate phase space, and we have determined the dimension of the model. [[25]] The problem is now to reconstruct the deterministic rule underlying the data. We start our discussion with the simplest and earliest nonlinear method of local forecasting, which was suggested by E. Lorenz (1969b). Let us propose to predict the values of $\boldsymbol{y}(k+1)$ knowing that a long time series of $\boldsymbol{y}(j)$ for $j \leq k$. In the '*method of analogs*' we find the nearest neighbor to the current value of $\boldsymbol{y}(k)$ say, $\boldsymbol{y}(m)$ and then assume that $\boldsymbol{y}(m+1)$ is the predicted value for $\boldsymbol{y}(k+1)$. This is pure persistence and is not much of a model." [26] To improve the quality of this prediction, Abarbanel et al. (1993) suggest to take a collection of near neighbors of the point $\boldsymbol{y}(k)$ and predict an averaged value of their images, and suggest to weight the neighbors to provide a larger contribution from close points.
- "Numerical results are critical to the study of nonlinear systems that have chaotic behavior. Indeed, computation plays a larger role in such studies than is traditional in many parts of the physics literature. Progress such as that reported throughout this review rests heavily on the ability to compute rapidly, and as such would not have been possible

---

[25] For this thesis, we do not need to concern ourselves with phase space and dimension determination problems because we: 1) have a large, representative record of a multidimensional time series, 2) know many of the relevant dimensions, 3) intend to straightforwardly perform analog forecasting (Lorenz 1969a), 4) do not intend to model the signal into uncharted space, and 5) do not intend to model the signal far into the future. We will simply collect an ensemble of analog trajectories and make reasonable inferences about the course and the predictability of short-term conditions. Nevertheless, the reader may be interested in the following technical chaos-related terms. A *phase space* is a coordinate space in which the coordinates are temporally related (e.g., x and dx/dt). Such a space can specify the state of a dynamical system. For example, simple harmonic motion is a circle in the phase plane. The *dimension* of the model is the number of coordinates needed to specify a state. When analyzing single scalar signals from systems with an unknown number of dimensions of freedom, a number supposedly greater than 1, some mathematical techniques, which are outside of the scope of this thesis, are necessary to estimate the dimension. For example, reconstructing the phase space through the technique of time delay embedding is a systematic way of transforming scalar data to a multidimensional phase space (Aberbanel et al. 1993).

[26] Pure persistence is not much of a model but it is effective nonetheless (see our *Additional references on analog forecasting in meteorology*). Interestingly, proponents of CBR often cite the model-free quality of CBR as an advantage. Models are difficult to construct in areas where domain theory is weak or domain experts are unclear.

a decade ago. The subject reviewed here is almost 'experimental' in that sense through its reliance on computers as an instrument for its study. This bodes well for further analysis of chaos and its physical manifestations , since no one can expect even more powerful computers to be available on a continuing basis."

- "It would be enormously useful, for example, in the analysis of data to have some estimate of the error in all the conclusions arising from finite data sets as well as from numerical, experimental, or instrumental uncertainty."
- "The discovery of temporal chaos in physical systems is a 'finished topic of research,' we believe. Many physical systems have been shown to exhibit chaotic orbits, and we are certain many more will be found. It is no longer enough, however, to have good chaos, one must move on and extract good physics out of it."

We intend to put "good physics" into the analog forecasting method by equipping fuzzy sets to measure importantly close physical dimensions.

To implement an analog forecasting method, the main problem to solve is to *find good analogs.* Somehow we must select past cases whose attributes are most similar to those of a new, partial, and (in the case of complex natural phenomena, such as weather) probably unique case.

1.5.3.4 Persistence climatology: Analog forecasting with built-in constraints

"Persistence climatology is widely recognized as a formidable benchmark for very short range prediction of ceiling and visibility [which are critical attributes of airport weather]" (Vislocky and Fritsch 1997).

Persistence climatology (PC) is a weather prediction technique that combines the best qualities of two basic weather prediction techniques: persistence forecasting and climatological forecasting. PC is a form of analog forecasting, and analog forecasting is meteorological version of CBR.

Huschke (1959) defines *persistence forecast* as: "a forecast that the future weather conditions will be the same as the present conditions." Huschke (1959) defines *climatological forecast* as: "A weather forecast based upon the climate of a region instead of upon the dynamic implications of the current weather. Consideration may be given to the climatic behavior of such synoptic weather features as cyclones and anticyclones, fronts, the jet stream, etc." The time of year (i.e., Julian day) is a condition that strongly determines the evolution of weather, on both the large scale and the local scale. [27]

---

[27] Barry and Chorley (1968) explain large-scale weather patterns correlate to particular dates:

> Recurrent weather spells about a particular date (singularities), such as the tendency for anticyclone weather in mid-September, have been recognized in Britain and major seasonal trends in occurrence of airflow regimes can be used to define five natural seasons [in Britain.] … Three major North American singularities concern the advent of

Martin (1972) explains how PC combines persistence and climatology to forecast cloud ceiling and visibility. We summarize his description as follows.

> The basic objective of PC is to answer the question: In similar past situations, what were the outcomes 1, 2, 3,... hours later? PC is a meteorological application of joint probability. For example, suppose that it is 6 am in June and the airport is "socked in" in fog. The flying category is the lowest possible, Category 1. Using PC, one tabulates before-the-fact probabilities (prior probabilities) to forecast for such a situation. The database is searched for all instances of {*June, 6 am, flying category 1*}, the flying categories during the subsequent hours are tabulated, and probabilities were prepared accordingly.

The elements of PC forecasting are shown in Figure 6. For purposes of illustration, the weather data are simplified. [28] Only four attributes are shown. In practical systems, many more attributes may be used. The *present case* is the incomplete present case that we want to predict for. The *predictands* are the missing parts of the present case, what we want to predict. The *past case* is an archived case which nearly matches the known attributes and auxiliary predictors of the present case. The *auxiliary predictors* are information about the present case available from sources other than direct observation, such as NWP or human estimation.

---

spring in early March, the midsummer northward displacement of the sub tropical high-pressure cell, and the Indian summer of September–October.

[28] The weather data are from weather observations (METAR code) for Halifax International Airport for the period from 00:00 to 12:00 UTC September 12, 1999.

	present	**case**			**past**	**case**		
time (UTC)	cloud ceiling (m)	visibility (m)	dew point temp. (°C)	wind direction (degrees) speed (kts)	cloud ceiling (m)	visibility (m)	dew point temp. (°C)	wind direction (degrees) speed (kts)
00h	210	2400	20	190° 10	90	2400	20	190° 12
01h	90	2400	20	200° 11	120	2400	20	200° 11
02h	120	12800	20	230° 11	120	3200	20	210° 10
03h	150	3200	20	230° 8	150	4800	20	220° 9
04h	210	16000	19	210° 9	210	6000	19	230° 8
05h	300	16000	18	320° 6	240	7200	18	320° 7
06h	240	19200	18	320° 5	300	19200	18	320° 6
07h	unlimited	24000	17	330° 5	4000	24000	17	330° 6
08h	7500	24000	16	330° 6	7500	24000	16	320° 6
09h	7500	16000	16	310° 7	7500	24000	16	310° 6
10h	7500	19200	15	290° 5	7500	24000	15	300° 6
11h	7500	19200	15	290° 9	7500	24000	15	290° 6
12h	7500	24000	14	290° 6	7500	24000	14	290° 6
	↑	↑	↑	↑ ↑				
		These two are **predictands**		These three may be anticipated with **auxiliary predictors**				In a forecast setting, grayed-out values are not known. They are objects of prediction.

Figure 6. Persistence climatology (PC) bases predictions for the present case on the outcomes of similar past cases. For example, two simplified series of hourly weather observations are listed. The present case is based on actual observations from Halifax International Airport (from September 12, 1999). The past case is a hypothetical analogous case. [29] PC makes predictions from such analogous past cases. In an actual forecast setting, the attributes of the present case would only be known up until 04h, whereas attributes of the analogous past case are known for the entire time span. Note that after 04h, in both cases case, the wind veers suddenly to northwest, dew point temperatures fall sharply, and low cloud clears quickly. Certain near-term attributes of the present case, such as dew point temperature and wind, may anticipated using auxiliary predictors from existing objective prediction techniques (e.g., numerical weather prediction).

Because analog forecasting is fundamentally different from NWP, it complements NWP. Therefore, analog forecasting has potential applicability for *postprocessing of NWP output*. Postprocessing of NWP output is the process of combining NWP output with complementary information and forecasting techniques. [30]

---

[29] We assume that the principle of analog forecasting is applicable and, therefore, assume that the weather archive, which is composed of 13,149 days of past cases, contains analogs for the present case. For

1.5.3.5 Fuzzy *k*-nn based forecasting: Analog forecasting without built-in constraints

The fuzzy *k*-nn technique can free persistence climatology (PC) from two of its main limitations and thus make PC more flexible and better able to take advantage of available data. The two limitations on the flexibility of previous PC systems are:

- Previous PC systems treat weather as if it was categorical, (and therefore)
- Previous PC systems can only use a very limited set of predictors.

One problem with representing weather cases according to the membership of those cases' attributes in crisp categories (as all previous PC systems do) is that such categories may not accurately reflect the level of similarity between cases, as illustrated in Figure 7.

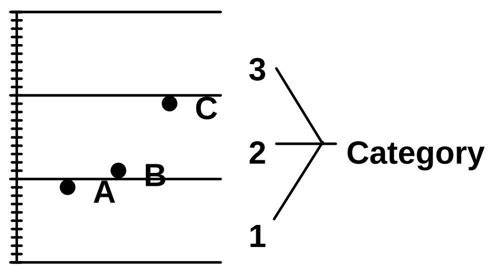


Figure 7. Crisp categories may not accurately reflect the level of similarity between cases. Such categorization may produce counterintuitive results. For example, the values of points A and B are similar and the values of points B and C are dissimilar, but points A and B fall into different categories and points B and C fall into the same category.

Another problem with using crisp categories to represent weather cases is that, as the number of stratifying conditions increases and as specified events become rarer, instances for

---

purposes of illustration, in Figure 6, we contrive a simple analog. However, in our experiments, we will only compare actual cases.

[30] Cats and Wolters (1996) describe postprocessing of NWP as follows: "Modern numerical weather forecasting systems have three basic components: an analysis unit, a forecast model, and a postprocessor. … In the postprocessing step, the relevant weather phenomena (for example wind speed at 10 m height are calculated from the model variables." In our case, the relevant weather phenomena are cloud ceiling height and horizontal visibility.

statistical tabulation may not exist. Martin (1972) attributes the problem to "rare events," but, to be more exact, the problem is that the more precise a crisp range query is, the greater the chance of finding no match. Therefore, previous PC systems have only used, or taken advantage of, a limited number of predictors so as not to produce "empty bins." [31]

All gardeners are familiar with crisp climate classification schemes, or "growing zones," and understand how such schemes are simplistic and potentially misleading. [32] McBratney and Moore (1985) applied fuzzy logic to the problem of climatic classification. From their results, they found:

> it appears the fuzzy sets approach has a useful place in climatic classification," [and suggest three reasons for the efficacy of fuzzy sets approach are that it] is realistic, flexible, and may offer better approach to information transfer than does the classification of climate into discrete sets.

We agree. McBratney and Moore (1985) emphasize that climate variables are continuous and that boundaries, if they exist, are fuzzy. They suggest that

> the apparent arbitrariness of [conventional, crisp] climate classification suggests an alternative approach would be the storage of climatic data in easily accessible form and the generation of a specific 'classification' for a particular purpose when it arises, using multivariate techniques.

Our fuzzy *k*-nn algorithm follows this suggestion to gear the algorithm for a particular purpose. It searches the stored weather observations and retrieves the *k*-nn which most belong to the momentarily most important class of weather observations, a class whose centre is defined by the features of the current weather situation, the latest series of airport weather observations.

We reviewed the meteorological literature on airport weather prediction systems and found only two systems that demonstrated accurate prediction results comparable to the

---

[31] This is reminiscent of the "accuracy-versus-precision" tradeoff in weather prediction. That is, the more precise a prediction is, the less chance there is that it will be accurate. In the case of data base querying, the more precise a query is, the less chance there is that a matching item will be found.

[32] In a gardening encyclopedia, Bradley and Ellis (1992) explain "growing zones" as follows: "In order to help growers determine which plants are best for their regions, the USDA's [US Department of Agriculture] Agricultural Research Service developed a Plant Hardiness Zone Map. … It divides the United States and southern Canada into 11 climatic zones, based on the average annual minimum temperature for each zone. Zone 1 is the coldest, most northerly region, and Zone 11 is the warmest, most southerly. … Keep in mind that there are climatic variations within each region and even within each garden. Your garden's immediate climate may be different from that of the region overall. Many factors—altitude, wind exposure, proximity to bodies of water, terrain, and shade—can cause variations in growing conditions by as much as two zones in either direction." Thus, for example, although general growing conditions in Nova Scotia are Zone 5, local growing conditions may range from Zone 3 to Zone 7.

benchmark prediction technique of persistence forecasting. [33] Both systems are based on multi-linear regression and are described as follows.

- Wilson and Sarrazin (1989) describe a refinement of PC called “SHORT” (the name is unexplained) that performs very well. SHORT describes the climatology of changes in aviation weather parameters based on 30 years of record. SHORT is more skillful than “conditional climatology” at all forecast ranges. SHORT is apparently still unrivalled by any other category-based PC system and is undergoing continued development.
- Vislocky and Fritsch (1997) describe a refinement of PC called “OBS” (the name is unexplained) that performs very well. What is special about OBS is that it incorporates observational weather data from *surrounding* airports, as well as from the particular airport in question, into the prediction process for the airport in question. Considerable skill is attributed to their unique inclusion of such highly relevant, predictive information. The authors suggest that further gains can be made in the future by somehow including more predictive information from other sources.

Despite their obvious skill, both systems have what we perceive to be design flaws which are inherent in all category-based and thus category-constrained PC systems to date. Wilson and Sarrazin (1989) explain, in SHORT, “all predictands and predictors are categorized,” and proceed to illuminate two problems arising from the use of categorization. First, there is not a single, consistent method to choose “best” categories. The “best” categories vary from one situation to another.

> Many strategies are available for choosing ‘best’ categories, the definition of ‘best’ depending on the use of the forecast. (Wilson and Sarrazin, 1989)

Second, categorization loses detailed information.

> The categorization procedure is considered necessary because the [multivariate linear regression] procedure produces large volumes of probability forecasts for each station and projection time. This procedure effectively summarizes the information but also makes the category decision for the forecaster and loses the detailed information available from probabilities. (Wilson and Sarrazin, 1989)

---

[33] There are numerous airport weather prediction systems described in the literature (e.g., Clarke 1995; Garner 1995; Gollvik and Olsson 1993; Keller et al. 1995; Kilpinen 1993; Kumar et al. 1994; Meyer 1995; Porter and Seaman 1995; Shakina et al. 1993; Warner and Stoelinga 1995; and Whiffen 1993), but none of these works claim to be as nearly as skillful, significant, or practical as the two referred to above, SHORT (Wilson and Sarrazin 1989) and OBS (Vislocky and Fritsch 1997).

When Wilson and Sarrazin (1989) speak of loss of detailed information, they refer only to loss at the *output* stage, but the same sort of loss occurs at the *input* stage. SHORT processes data in three stages:

1. It receives many detailed cases and converts them into categories, thus discarding detailed information about similarity between weather cases at the input stage.
2. It develops regression equations based on categorical representations of cases.
3. It outputs the results of those operations into categorical predictions, thus losing the detailed information available from probabilities calculated during stage 2.

Wilson and Sarrazin (1989) recommend trying new graphical output procedures to recover some of the information lost at the output stage. But the detailed information about similarity between cases lost at the input stage to multivariate linear regression due to categorization is irretrievable. Details that could enable the measurement of *level of similarity* of analogous cases are discarded through categorization.

Another problem inherent in both SHORT (Wilson and Sarrazin 1989) and OBS (Vislocky and Fritsch 1997) is that they do not incorporate into the prediction scheme numerous real-time predictors, such as data from surrounding weather-measuring stations or from upper air stations. There is a practical limit on the number of combinations of attributes that statistics can be prepared for. The size of the equation set tends to grow exponentially with the inclusion of each new attribute.

In contrast, the fuzzy *k*-nn system grows linearly in complexity as new attributes are added (illustrated in Figure 9, page 64). Thus, it can potentially take better advantage of many valuable real-time predictors—variable, situation-specific predictors that are relevant to current weather. Operationally, forecasters know more than what month it is and what time of day it is. They have additional knowledge about what the “problem of the day” is. For example, if a cold front is due to pass through the region during a forecaster’s shift, then all weather timings hinge on the time of passage of a cold front. “Timing the cold front” and its associated wind shift is the most critical task for an aviation forecaster on such a day.

Timing of fog and ceiling lifting often depends on the passage of a cold front. For instance, forecasters may determine, either manually or automatically, that wind direction will shift from 160° to 320° three hours after forecast time. Fuzzy *k*-nn analog forecasting can begin with that information. It searches the record of over 300,000 consecutive hourly airport reports for the few most similar situations in the past, similar according to all the commonly known attributes *plus* the very predictive information about wind direction shift three hours hence. Those few analogs are most relevant and excellent for prediction.

Presently, with PC, there is not an easy way to specify such a peculiar set of conditions as {*June, 6 am, 1/4SM FG, OVC001, wind shift from 160° to 320° three hours hence*}, because PC must be prepared before-the-fact, using only a limited set of commonly used predictors. Information about such peculiar cases, contained in the database, is not presently made available to forecasters. It is impractical to prepare PC statistics for the full range of possible situations and relevant predictors. Whereas, the fuzzy *k*-nn technique can select *the k*-nearest neighbors, nearest in terms of a set of critical attributes (i.e., predictors) which can be known only at forecast time.

From a user's perspective, fuzzy *k*-nn is flexible. A meteorologist colleague of ours described it as "custom climatology on-the-fly." A system can defer important decisions until run-time. A forecaster invests considerable effort in timing a cold front. The forecaster could use "real-time persistence climatology" simply by entering the precise expected wind-shift attributes into a system, let the system automatically supply the other predictors (e.g., time of day, month, surface observations, NWP), and have the system output the likeliest trend of ceiling and visibility using all the available data and the best analogs.

To the best of our knowledge, "pure analog forecasting" has never before been used to produce airport weather forecasts. By pure analog forecasting, we mean making forecasts based on a *few actual most similar cases,* similar according to the salient attributes of the present case and selected from an entire archive, rather than making forecasts based on statistically derived probabilities, probabilities that are determined according to general attributes of many cases.

All work to date to automate airport weather prediction has used some combination of all of the three methods: climatology, numerical weather prediction (NWP), and statistics. Climatology describes the past behavior of specific weather conditions at an airport. NWP provides guidance about near-term future conditions. Statistics let us calculate conditional probabilities. A system combining these three methods is often referred to as model output statistics (MOS). [34] All such MOS work is based on two assumptions:

---

[34] Strictly speaking, "MOS" has a more restricted meaning in meteorology. Depending upon the approach towards climatology, NWP, and statistics, a hybrid method may be referred to differently, e.g., "persistence climatology" (Martin 1972), "SHORT" (Wilson and Sarrazin 1989), "OBS" (Vislocky and Fritsch 1997), and "perfect prog" (Stern and Parkyn 1999) to name just a few. But such meteorological semantics are beside the point. The point is that all serious attempts to automate airport weather prediction:

- Combine three basic prediction methods: climatology, NWP, and statistics.

1. *Analog forecasting*: good analogs make good predictions. Similar weather situations, patterns and sequences, behave similarly.
2. Similarity can be adequately described according to membership in a few, arbitrary, crisp, predefined categories.

We will use the first assumption because we believe it's a principle with wide applicability. However, we will not use the second assumption. We claim that it is a compromise, an oversimplification of data, that has become increasingly unnecessary. In the past, with relatively limited computing power, to process large weather databases, it was necessary to greatly condense them, to preprocess them before the reception of up-to-the-minute predictive information. The fuzzy *k*-nn method can process large databases efficiently after the receiving the specific details of a new case and thereby perform unconstrained analog forecasting.

1.5.3.6 Fuzzy *k*-nn algorithm's improvement to analog forecasting

Our fuzzy *k*-nn algorithm can improve analog forecasting because the flow of weather is sensitively dependent on initial conditions. The fuzzy *k*-nn identifies the most similar cases regardless of categories. Previous analog forecasting techniques used predefined categories and thus must have failed to measure sensitively dependent conditions.

Analog forecasting of aviation weather using the fuzzy *k*-nn algorithm is more flexible and, thus, potentially more useful than previous category-based analog forecasting systems.

Previous analog forecasting systems: assume a limited number of predictors, represent attributes of cases according to their membership in categories, and prepare probabilities of categorical events accordingly. Whereas fuzzy *k*-nn based analog forecasting can: use any predictors which are available at run-time, represent attributes of cases with their full measured precision (thereby preserving information that improves similarity measurement), and prepare analog predictions based on a few individually-weighted, most-similar temporal cases.

- Implicitly use the analog forecasting principle. For matching analogs, climatology gives hindsight about attributes of past analogous cases and NWP gives foresight about certain attributes of the present case. Statistics categorizes such attributes of past and present cases and makes inferences, or predictions.
- And, prior to this thesis, implicitly assume that similarity, of new and unique weather cases with old weather cases, can be adequately described according to membership in a few, arbitrary, crisp, predefined, mediating categories, rather than by direct case-to-case comparison.

Using the fuzzy *k*-nn method frees us from the dependence on the assumption (implicit in all previous attempts to automate airport weather prediction) that similarity can be adequately described according to membership of case attributes in a few, arbitrary, crisp, predefined categories. The consequence of categorization of case attributes—when categories are defined for general situations, without regard to specific cases or case-specific contextual information—is that precision of distance measurement between analogous cases is reduced. Categorization is "lossy," so to speak. However, attempts to circumvent the problem of lossy categorization by creating more and finer categories simply leads to another problem: the increased chance that there will be too few instances to base probabilities upon. In contrast, the fuzzy *k*-nn method avoids categorization and both of its related problems by, in the sense of (Viot 1993), fuzzifying input and defuzzying output.

To the best of our knowledge, the fuzzy *k*-nn technique described in this thesis is the only example of the use of a proper *distance function* and *metric space* being used for airport weather prediction. This is further explained in Chapter 2. [35]

In Chapter 2, we survey how others have used fuzzy logic for retrieval. In Chapter 3, we describe our implementation of a fuzzy *k*-nn algorithm for airport weather prediction.

---

[35] For definitions of proper distance function and metric space, see footnote number 42 ahead on page 52.

2. Literature Survey

In the previous chapter, we explained how retrieval of similar cases relates to CBR, fuzzy logic, and weather prediction; namely: CBR depends on retrieval of similar cases, fuzzy logic enables retrieval of similar cases, and the weather prediction technique of analog forecasting depends on retrieval of similar cases. We also explained why case adaptation and case authoring are regarded as main challenges in CBR system development.

In this chapter, we survey the literature to explain how using a fuzzy k-nearest neighbors based technique for retrieval of similar cases, designed and tuned with the help of domain expert, can help us to exploit large databases of cases and available domain knowledge about similarity, and can help us to avoid difficulties of case adaptation and case authoring.

In section 2.1, we describe the main resources for CBR. In section 2.2 we review how fuzzy logic is used in CBR. In section 2.3, we provide a foundation for the fuzzy k-nearest neighbors (fuzzy *k*-nn) technique. In section 2.4, we review a number of CBR applications that exemplify the fuzzy *k*-nn technique. In section 2.5, we review weather prediction papers that use CBR and fuzzy logic.

2.1 Resources for case-based reasoning

The main resources for case-based reasoning are (of course): cases, a method for reasoning, and software. The points of this section are: CBR scales up to take advantage of large databases of cases, one can reason on the basis of similarity alone, and domain knowledge improves the process of determination of similarity, and existing software may or may not be helpful.

2.1.1 Large databases of cases

CBR scales up to take advantage of large databases of cases . Creecy et al. (1992) describe an early successful large scale *k*-nn system, called "PACE." For the 1990 United States Census, 22 million natural language census returns had to be classified into 232 industry categories and 504 occupation categories. The case base consisted of 132,000 previously classified returns.

Before PACE, census classification required expensive, labor-intensive clerical work. An expert system, called "AIOCS," was developed in 1990 to assist clerks. PACE required 4 person-months to be built, whereas AIOCS required 192 person months. PACE successfully processed 60% of returns whereas AIOCS processed 47%.

The larger the database and the denser the examples close to the case, the better the accuracy of PACE's performance.

Confidence measures were a byproduct of the *k*-nn approach. PACE generated a nearness measure for each example. If any new example was identical or very similar to a previously seen case, then PACE attached high confidence to its results. If no previously seen case matched closely, PACE reported the closest cases and, additionally, informed the user that the results were dubious.

Gentner and Forbus (1991) describe a model of similarity-based retrieval called MAC/FAC, short for "Many Are Collected, Few Are Chosen." The idea is to exploit large case bases with minimal computational cost. Many potentially similar cases are screened using a simple test, then a few probably similar cases are ranked for similarity using a more detailed test. The MAC part encodes structured representations as *content vectors* whose dot product yields an estimate of how well the corresponding structural similarities will match. The FAC part performs a more detailed, computationally expensive structural mapping. MAC/FAC inspired many researchers.

Scaling up helps us to avoid the case adaptation problem of CBR. The more cases we evaluate, the better the chance that good analogs exist and that there is less requirement for adaptation.

2.1.2 Domain knowledge about similarity

One can reason on the basis of similarity alone, and domain knowledge improves the process of determination of similarity. Cain et al. (1991) use domain knowledge to influence similarity judgement. A few very simple domain-based rules about which combinations of attributes are more important than others significantly improves CBR performance.

Aamodt and Plaza (1994) identify a trend in CBR research and development. Whereas pioneering work stressed the cognitive science based view of CBR as a plausible, general model of intelligence, more recent work emphasizes the importance of knowledge acquisition in the development CBR systems. Based on this trend, Aamodt and Plaza (1994) urge CBR researchers to aim for "flexible user control [and] total interactiveness of systems." CBR becomes more applicable as it integrates knowledge based techniques. Of particular relevance for this thesis, they explain:

> The ‘indexing problem’ is a central and much focussed problem in case-based reasoning. It amounts to deciding what types of indexes to use for future retrieval, and how to structure the search space of indexes. Direct indexes [i.e., examining surface features] skips the latter step, but there is still the problem of identifying what types of indexes to use. This is actually a knowledge acquisition problem…

Watson and Marir (1994) survey CBR research and conclude:

> Case-based reasoning will be ready for large-scale problems only when retrieval algorithms are efficient in handling thousands of cases. Unlike database searches that target a specific value in a record, retrieval of cases from the case base must be equipped with heuristics that perform partial matches, since in general there is no existing case that exactly matches the new case.

The fuzzy *k*-nn method acquires domain-based knowledge about how to perform partial matching and this acquired knowledge guides retrieval of similar cases from large case base (Hansen and Riordan 1998). Reasoning on the basis of similarity helps us to avoid the case authoring problem: using similar, automatically-recorded cases eliminates the dependency on having a person handcraft cases.

2.1.3 Commercial software tools

Hashemi (1999) reviewed numerous recently developed CBR software tools. [36] The tools are described in terms of how they assist system development and how the systems operate. In general, the tools share the following properties: For development, the systems help people to construct case libraries. Libraries are constructed through hierarchical organization of existing cases and through helping people to author new problem-solving cases. For operation, the systems help users to retrieve similar and analogous cases. Retrieval uses rules, object hierarchies (i.e., decision trees), and nearest-neighbor algorithms. Each of the reviewed tools is a sort of *production system* [37] where, in the context of CBR, the productions are problem-solving cases in a library.

---

[36] The review by Hashemi (1999) describes eleven CBR software tools: Case-1 by Astea International, ART*Enterprise and CBR-2 (CBR Express, CasePoint, Generator, and Tester) by Inference Corporation, CasePower by Inductive Systems Inc., Eclipse by Haley Enterprises, ESTEEM by Esteem Software Inc., KATE by Acknosoft, ReCall by Isoft, ReMind by Cognitive Systems, S3-Case and CBR Works-4 by TechInno.

[37] A *production system* consists of a collection of productions (rules), a working memory of facts and an algorithm known as forward chaining for producing new facts from old. A rule becomes eligible to “fire” when its conditions match some set of elements currently in working memory. A conflict resolution strategy determines which of several eligible rules (the conflict set) fires next. A condition is a list of symbols which represent constants, which must be matched exactly; variables which bind to the thing they match and "<> symbol" which matches a field not equal to symbol. (Definition from the Free Online Dictionary of Computing, http://wombat.doc.ic.ac.uk/foldoc, downloaded Oct. 27, 1999).

CBR software tools commonly assume that cases divide into hierarchies. For example, early applications dealt with dinner-planning: meats divide into classes such as chicken, fish, and so on (Riesbeck and Schank 1989). For another example, an infection-diagnosing system could classify germs as either “gram positive” or “gram negative” (using a test in which a dye is applied to a sample and the sample is examined to see whether the dye was absorbed). Weather cases do not divide into hierarchies, or distinct classes. For any two distinct weather cases, a third case could occur which would be midway between the first two. If the first two cases are used to define hierarchies, or classes, then classification of the third case would be ambiguous.

It can be difficult to adapt available CBR software tools for specific applications. [38] So rather than using such a tool, we chose to develop a unique fuzzy *k*-nn system using the C programming language. Our three main reasons for building a system from basic components are:

- *Data opportunity*. The fuzzy *k*-nn weather forecasting system is not dependent on a library of prototypical cases. It does not require manual or automatic construction of a library. Instead, it uses existing, huge archives of ready-to-use weather data. Moreover, these archives are continuously growing as new airport weather observations are made.

---

[38] “At the risk of being ostracized by the CBR community,” Laight posted the following list of CBR downsides to the AI-CBR mailing list in response to a request for downsides, on July 20, 1999. The list, informally posted by Graham Laight to the AI-CBR E-mail list, is the nearest thing to a consensus that we have found so far—none of the many CBR-savvy recipients of Laight’s e-mail challenged this list. (“There are currently 660 registered members (as of May 1999) [of the AI -CBR mailing list],” according to Ian Watson, the maintainer of the list (membership number downloaded from http://www.ai-cbr.org/theindex.html on Sept. 25, 1999.)

CBR downsides

- High cost of CBR tools.
- Absence of a "standard" CBR tool, in the sense that MS Word is word processing standard.
- Cost and difficulty of implementing a CBR system.
- Difficulty of changing, or adapting, a system once it is implemented.
- You may find it just as easy (and successful) to build your application with a normal database package (or even a spreadsheet!), unless your requirements make a good fit with a CBR package. You may even find a document search type application meets your needs.
- Cost of maintaining the case base, especially if the scope of the case base changes in any way.
- Information in the case base may become outdated.
- Each case may have to be carefully prepared by an expert.
- Depending on the application, it may be possible to get the information [relevant for problem the application is meant for] in other ways.

- *Data integrability*. An advantage of applying fuzzy logic to data mining is that it will enable us to integrate new forms of data with records of old data. For future work, we plan to combine satellite data and airport weather observations. Operationally, satellite data is one of the most valuable forms of predictive guidance for aviation forecasting, particularly short-term projection of satellite images. The archive of hourly airport surface weather observations, which stretches back 36 years, does not contain accompanying satellite images, but in many instances we can reasonably infer what satellite images would have looked like based on the surface observations. For example, if surface conditions change from clear skies to rain, we can reasonably infer that satellite images would have shown a change from clear to overcast. Past cases can be augmented with such inferred attributes, what would have been shown with satellite data had it existed, and then compared with present cases which have actual satellite data. Thus, we can exploit short-range projections of satellite data to search the archive more intelligently.
- *System flexibility*. The available weather data and related weather forecasting systems and programs depend mainly on C programs, so we will be able to integrate our system effectively by developing parallel, easily modified code.

We are encouraged by the example of Baldwin and Martin (1995) who applied fuzzy logic to data mining (they call their tool a "fuzzy data browser") and found fuzzy logic to be advantageous in three ways.

- *Finding relationships.* Their system develops fuzzy rules [39] that describe relationships within the data, based on numerous cases, and can thus evaluate new cases accordingly: "Each rule corresponds to a summary of several 'similar' cases in the database into a fuzzy prototypical rule. The value of a new case then corresponds to an interpolation between these fuzzy prototypes." An extension of this is "that it is possible to use the rules to highlight anomalous values." New cases that do not fit previous patterns indicate new relationships that may warrant scrutiny.
- *Involving and exploiting intelligent users.* "The system can run entirely autonomously, or the user can inject expertise either in … suggesting attributes

[39] Baldwin and Martin (1995) regard fuzzy rules which condense data and "fuzzy prototypical cases" as synonymous. An example of a fuzzy rules is "*if A and B then C*," where *A*, *B*, and *C* are fuzzy sets.

which should be target for prediction … or suggesting attributes which should be used to make the prediction … or suggesting compound features. … The browser can be used to explore intuition about the underlying relationships in the data, or left to run autonomously and discover relations that can be presented to the user."

- *Combining new forms of data with records of old data.* "In many cases, data may not be complete—for example, weather records could include daily temperature extremes, rainfall, cloud cover, etc. If a new machine became available for monitoring atmospheric pollution, these measurements could be added to the database, but the earlier records would not contain this information. If there is a relation between atmospheric pollution and some of the other recorded quantities, rules modeling this relationship could be used to make an intelligent guess as to the pollution level for each record where it was not measured." This *data integrability* advantage of fuzzy logic will enable us to integrate into new cases new forms of data, such as satellite data, and compare such enriched present cases (composite cases) with past cases.

In this section, we described how, to overcome the case adaptation and case authoring problems of CBR system development (which are described in subsection 1.3.2), three strategies have been proposed: knowledge acquisition, partial matching, and use of very large case bases. The rest of this chapter surveys papers that apply fuzzy logic to use these strategies.

2.2 Fuzzy logic and case-based reasoning

López de Mántaras and Plaza (1997) surveyed over 100 recent CBR papers and concluded that

> the use of Fuzzy Logic techniques may be relevant in case representation to allow for imprecise and uncertain values in features, [and] case retrieval by means of fuzzy matching techniques. [Moreover] perhaps the most severe limitation of existing systems [all types of CBR systems] is the feature-value representation that is being used for cases. The consequence is that case-based algorithms cannot be applied to knowledge-rich applications that require much more complex case representations, for example cases with higher-order relations between features.

The CBR process of *matching cases* can be carried out by the fuzzy logic process of *measuring the degree of similarity of cases*.

CBR and fuzzy logic both deal with how to determine *degree of similarity*, but they tend to use different approaches. CBR commonly deals with features, geometry, and structure (Bridge

1998, and Liao et al. 1998), whereas fuzzy logic deals explicitly with uncertainty and ambiguity expressed intentionally by humans when they are asked to describe similarity. Fuzzy words describe uncertainty intentionally and fuzzy sets represent the intended uncertainty.

2.2.1 Fuzzy CBR formalism

Fuzzy CBR is a type of CBR that uses fuzzy methods to represent and compare cases, and to form solutions. The implicit principle of the fuzzy *k*-nn method is expressed by Dubois et al. (1997) as: “the more similar are the problem description attributes, the more similar are the outcome attributes.”

Their paper is relevant for this thesis for two reasons. First, for foundation, it provides a general mathematical formalism with which to take advantage of this principle. The fuzzy *k*-nn method applied to a weather prediction problem is an actualization of this general formalism. In this subsection, we summarize their formalism in order to provide a basis for the fuzzy *k*-nn method. Neither we nor Dubois et al. (1997) investigate learning aspects of CBR.

Second, it describes how a fuzzy set framework accommodates two types of problems: deterministic and non-deterministic. As we noted in the Introduction, the weather prediction problem, which is deterministic in theory, is non-deterministic in practice because of uncertainty surrounding weather measurements. Fuzzy sets represent this uncertainty.

Our interpretation of fuzzy CBR formalism is consistent with that given by Dubois et al. (1997). This subsection reviews and summarizes their description, nearly verbatim.

A *case* is viewed as an *n*-tuple of precise attributes. The set of attributes is divided into two non-empty disjoint sets: the problem description attributes subset and the solution description attributes subset, which are denoted by S and T respectively.

A case is denoted by a tuple (s, t) where s and t stand for complete sets of precise attribute values of S and T respectively.

To perform case-based reasoning, we relate problems with solutions. We assume that we have a finite set M of stored cases called a *case base* or *memory* (M is thus a set of pairs (s, t)), and a current problem description denoted by s_0, for which the precise values of all attributes belonging to S are known. Then case-based reasoning aims to extrapolate an estimate of the value t_0 of the attributes in T for the current problem.

In this setting, the implicit case-based reasoning principle is defined as:

> The more similar are the problem description attributes, the more similar are the outcome attributes.

This is the implicit principle that underlies all deterministic CBR. If two items are perfectly similar then they are identical. So in theory, if a problem is well-posed—if a problem-to-solution mapping is many-to-one or one-to-one—then the solution is deterministic. In a deterministic setting, the CBR principle may be expressed with the following constraint

$\forall(s_1, t_1), (s_2, t_2) \in M, S(s_1, s_2) \leq T(t_1, t_2)$

This means that the similarity of s_1 and s_2 constrains the similarity of t_1 and t_2 at a minimum level. In other words, the problem is well posed. For deterministic CBR to be applicable, cases must map to the solution space in a many-to-one or a one-to-one way. Deterministic CBR is inapplicable when cases map to the solution space in a many-to-many way.

Expressed in terms of fuzzy relations on S and T, the implicit CBR principle can be expressed with the following rule:

The more S-similar s_1 and s_2, the more T-similar t_1 and t_2

where $(s_1,t_1,), (s_2, t_2) \in M$. A problem in this fuzzy CBR framework is denoted with the 4-tuple $\langle M, S, T, s_0\rangle$ where the above principle is applicable.

All of the applications described in this chapter (ahead in Section 2.4) and our own application (in Chapter 3) exemplify of the use of this principle. A detailed example of the use of this principle is given in the Appendix B: *A Worked-out Example of Fuzzy k-nn Algorithm for Prediction*.

In reality, however, problems are often ill-posed. [40] In the physical world, attributes are rarely known precisely and with certainty. This makes efforts to develop deterministic CBR futile. This is also why very long-range weather predication remains impossible.

> Prediction of the sufficiently distant future is impossible by any method, unless the present conditions are known exactly. In view of the inevitable inaccuracy and incompleteness of weather observations, precise very-long range forecasting would seem to be non-existent [and this conclusion does] not depend on whether the atmosphere is deterministic. (Lorenz 1963)

Long-range prediction of many real, dynamical systems is hampered in this way. Given enough time, chaos defeats determinism in natural systems across all scales, ranging from molecular to astronomical. [41]

---

[40] The snowboard analogy in Figure 5 illustrates how many-to-one mapping switches to many-to-many mapping over time.

[41] Simple three-body systems can have complex orbits, depending on the initial conditions (Lorenz 1993).

To deal with non-deterministic problems, Dubois et al. (1997) suggest to relax the constraints of the above-described formalism and to accept that "case-based reasoning can only lead to cautious conclusions." Indeed, when attempting long-range prediction using realistic, dynamical, analogous temporal cases, we should always qualify our conclusions. We should recognize the margin of error. Examining ensembles of cases, such as those shown in the snowboard analogy (Figure 5, page 29), enables us to determine appropriate margins of error to qualify our results.

To relax the constraints, Dubois et al. (1997) reword the principle slightly and use fuzzy sets to modify the constraint.

> the more similar s_1 and s_2, the more possible t_1 and t_2 are similar

Yager (1997) also argues for a unified view of fuzzy set theory and CBR, and goes so far as to contend that: "The reasoning process used in FSM [fuzzy systems modeling] and CBR are the same." The main distinction between fuzzy modeling and CBR identified by Yager is that

> fuzzy modeling is generally used in environments in which the required solutions are numeric values, whereas the case-based methodology has a more ambitious agenda regarding the domain of the possible solution. This more ambitious agenda comes at the price of not always having available the necessary operations to combine solutions.

Thus fuzzy CBR is a subset of CBR, a type of CBR that uses fuzzy methods to represent and compare cases, and to form solutions

2.2.2 Numerous conditions and partial matching

Fuzzy CBR combines numerous conditions and partial matching in queries. As the number of conditions specified increases, the chance of turning up a good match increases. The same principle underlies a Bayesian case-matching scheme described by Kontkanen et al. (1998). The basic idea in their scheme is to use prior and posterior probabilities of certain cases to adapt cases retrieved from a case base. Prior probability describes the likelihood of a feature *prior* to the query. Posterior probability describes the likelihood of a feature *after* the query. For example, at Halifax the prior probability of rain at any given hour may be near 10%, but the posterior probability of rain an hour after rain was observed may be near 90%. Kontkanen et al. (1998) explain how to use such information to weight cases appropriately when making solutions. But both types of probability have their problems. Using prior probability turns up many cases of low similarity, where as posterior probability turns up few cases, or no cases, of high similarity. Kontkanen et al. (1998) explain the necessity of having a "soft" metric for dealing with cases that may or may not have good matches in the case base. Their proposed future work is to further develop and test a “soft constraint approach.”

Their system uses cases that are continuous feature vectors which, as they note, are the commonest type of vector in CBR prediction research. They further explain how CBR distance metrics are themselves, in essence, restricting assumptions on the problem domain:

> in many traditional CBR systems, the algorithms typically use a distance function (e.g., Euclidean distance) for the feature vectors in order to determine the most relevant data items for the task in question. The use of a specific distance function implicitly assumes that the distance function is relevant with respect to the problem domain probability distribution, and hence restricts the set of distributions considered.

One commonly used simplifying assumption is that all the attributes are independent. This makes integration of probabilities simple and feasible. Each attribute brings its own independent probability distribution into the equation. They formulate a Bayesian similarity metric which exploits posterior probability.

Their experiment was to repeatedly select cases at random from the case base, remove some features, introduce small random modifications to the rest of the features of the case, and use the resulting partial scrambled case as a query on the case base. They found that the Bayesian similarity score produced better results than a simple Hamming distance similarity score. Original cases could be recognized in the case base based on only a few perturbed original attributes.

As they note in their conclusion, basic “Bayesian probability theory can be used as a formalization for the intuitively appealing CBR paradigm.” They cite one of the advantages of the Bayesian scheme is that it forces one to explicitly recognize all the assumptions made about the problem domain. Simple distance metrics can conceal assumptions about dependence or independence of features, assumptions that may or may not be correct.

Their conclusion is reasonable and hardly contentious. Basically they conclude that probabilistic methods, carefully used, are useful for CBR. We contend that fuzzy *k*-nn offers similar opportunities to complete partial cases by querying databases. Moreover, the fuzzy *k*-nn enables us to explicitly express assumed knowledge of similarity and assumptions about dependence or independence of features acquired from a domain expert. Assumptions about dependence and independence of features are determined by how fuzzy sets are operated upon. For example, the similarity of two case’s winds can be computed as a single value based upon the similarity of two interdependent wind variables (direction and speed), while the similarity of two case's humidities can be computed as another single value based upon the independent variable of humidity.

Bosc and Pivert (1992) explain, in formal mathematical terms, how fuzzy sets enable flexible querying of databases. Fuzzy sets enable imprecise queries on a database. Two situations in which imprecise query conditions are useful:

- When the user is imprecise, fuzzy sets model the imprecision.
- When a prespecified number of responses is desired, discriminating margins of fuzzy sets enable elements in the database to be ranked according to degree of similarity.

Bosc and Pivert (1992) suggest, for a research topic, "improvement of the discriminating capability of the fuzzy sets based approach in two extreme cases: no element is selected (null degree) and a too large number of elements which have received a degree equal to 1."

Bosc and Pivert (1992) used trapezoidal fuzzy sets. Such fuzzy sets under-utilize the discriminating power of fuzzy sets. Two elements whose memberships both equal 0 or both equal 1 are seen as equivalent even though they may differ slightly. However: unimodal fuzzy sets discriminate more effectively than trapezoidal fuzzy sets. In Chapter 3, we show how the problem of having too few or too many matches is avoidable through appropriate fuzzy set design. Non-zero, continuously-varying, unimodal fuzzy sets discriminate continuously between cases and, thereby, equip a similarity measuring function with the properties of a formal metric space. [42]

2.2.3 Flexible similarity-measuring framework

Fuzzy set theory is a flexible framework for measuring similarity that can: 1) include or exclude the properties of reflexivity, symmetry, monotonicity and transitivity; and 2) subsume

---

[42] The critical part of the fuzzy *k*-nn technique, which we describe in Chapter 3, is a similarity measuring function called *sim*. The complement of similarity is dissimilarity (1.0 - *sim*) and this dissimilarity operates like a distance function in a formal metric space. Kasriel (1971) defines a distance function and metric space as follows:

> Let d be a nonnegative real-valued function defined on $X \times X$ that satisfies the following: For all x, y and z in X,
>
> (a) $d(x, y) = 0$ if and only if $x = y$
> (b) $d(x, y) = d(y, x)$
> (c) $d(x, y) + d(y, z) \geq d(x, z)$ (triangle inequality).
>
> The function d is called a distance function or a metric for X and (X, d) is called a metric space.

When one uses trapezoidal fuzzy sets, property (a) may be violated.

both relative and absolute measures of similarity. These are commonly desired properties of similarity measuring functions in CBR.

In a description of a lattice-valued approach [43] to making similarity-measuring functions, Bridge (1998) identifies four properties that must be addressed in the design of similarity-measuring functions.

- *reflexivity:* $x \sim x \Rightarrow$ "topmost" similarity [44]
- *monotonicity*: continuously increasing or decreasing
- *symmetry:* $x \sim y = y \sim x$
- *transitivity*: $x \sim y$ and $y \sim z$ will imply $x \sim z$

Designers may or may not want to impose the above conditions. Reflexivity is usually a desired property and monotonicity is often desired. Symmetry may or may not be desired, depending on the intent. For example, we would assert that *rain* ~ *snow* = *snow* ~ *rain*, but *precipitation* ~ *snow* < *snow* ~ *precipitation*. [45]

Transitivity is usually not desired. For instance, knowing the values of *fog* ~ *rain* and *rain* ~ *snow* does not necessarily determine the value of *fog* ~ *snow* because the relationship between fog and snow is special and does not necessarily involve rain at all.

Bridge (1998) claims that the lattice-valued approach to measuring similarity is advantageous because it enables us to address all the four properties simultaneously. Similarly, we claim that fuzzy set theory is advantageous because, as a formalism, it accommodates variations of the four properties (Zimmerman 1991).

Bridge (1998) describes two basic types of similarity functions, absolute and relative. An absolute similarity function takes two representations and returns a Boolean result, either *similar* or *not similar* ($f_{Boolean}(x, y) = \{0, 1\}$), whereas a relative similarity function returns a number ($f_{Relative}(x, y) = \{0, 1\} = \mathbb{R}$). As Bridge explains, the problem with absolutism is that it

---

[43] The lattice is a graphical way of designing a similarity measure in which the edges of the graph correspond to similarity-measuring operations. Bridge (1998) defines a lattice as a "partially ordered set of values that satisfies certain properties." The set members are similarity-describing variables which may be in different forms, such as Booleans, numbers, or "hedge words" (e.g., {*very*, *quite*, *fairly*, *barely*}).
[44] The symbol "~" represents the similarity-measuring function, as in (Bridge, 1998). We understand $x \sim y$ to mean "the similarity of x to y." By "topmost," Bridge means the highest possible value of similarity. In a normalized fuzzy set, this equates to 1.0.
[45] If asked, "Is precipitation snow?" we would say maybe. If asked, "Is snow precipitation?" we would say yes.

does not correspond with "people's intuitive concept of similarity, in which there is a notion of 'degrees of similarity.'" And the problem with relativism is that

> on occasion, the numbers used are arbitrary. This occurs when similarity function designers need something richer than absolute similarity, i.e., they need degrees of freedom, but they do not need to quantify, or cannot properly quantify, the degrees. Any number used in these circumstances will be contrived.

Fuzzy set theory gives CBR system designers and CBR knowledge acquirers a full range of similarity-measuring functionality. The sets enable similarity-measuring results to be qualified or quantified, depending on whether the results are fuzzified of defuzzified, or results can simply be ranked according to degree of similarity.

Bridge (1998) claims versatility as another advantage of the lattice-valued framework:

> We claim that the advantages of our [lattice-valued] metric framework are that: it subsumes absolute and relative measures (i.e., these are instantiations of the framework); it introduces (again as instantiations) many other ways of measuring similarity [such as hedge words] (only a few of which other researchers have reported in the literature); and it allows the easy combination of similarity functions.

Fuzzy set theory also subsumes absolute and relative measures. Absolute measures are achieved by prescribing crisp sets, which are a subset of fuzzy sets. Relative measures are achieved by prescribing graded membership. One of the main recommendations of fuzzy set theory is its facility for operating directly with "hedge" words (e.g., Zadeh 1999, Zimmerman 1991, Cox 1992, and Maner and Joyce 1997).

Common types of variables used to describe features in case-based systems are continuous, ordinal, nominal, and interval-specific. Fuzzy set theory complements CBR by enabling us to represent features in another way, as Main et al. (1996) explain:

> A large number of the features that characterize cases frequently consist of linguistic variables which are best represented using fuzzy feature vectors.

2.2.4 Numbers and words

Jeng and Liang (1995) propose a fuzzy logic based approach to retrieval and cite three main advantages of the approach. First, it allows numerical features to be converted into fuzzy terms to simplify the matching process. Most existing CBR methods assume qualitative features, such as "weak or powerful weapon," but provide few options for how to deal with numerical features. Second, it allows multiple indexing of a single case based on a single attribute with different degrees of membership. A single case may be applied in various contexts—thus, it supports the relational database model as opposed to the structured database model. And third, it allows greater flexibility in the retrieval of candidate cases. Queries can be modified qualitatively and linguistically to suit special circumstances. For instance, one may request cases describing items that are *old* or in a smaller subset of *very old*, depending on the circumstances.

Jeng and Liang (1995) explain how the α-cut applies to CBR. The α-cut describes the subset of cases that have membership above a prescribed threshold. For instance, similar cases may have membership values in the range [0.0…1.0]. If we specify $\alpha = 0.5$, then the α-cut of similar cases will be only the cases with membership values, μ, such that $\mu \geq 0.5$. As Jeng and Liang (1995) point out, "delicate tradeoffs are involved in choosing proper α-cuts." For instance, choosing a high level of α increases efficiency by allowing the system to rule out cases with low membership but this tactic may also limit the retrieval flexibility of the system. We will use the α-cut convention when we describe our fuzzy *k*-nn weather prediction system in subsection 3.2.

2.3 Foundation for fuzzy *k*-nearest neighbors technique

This section provides a foundation for the fuzzy *k*-nn technique. Subsection 2.3.1 reviews the more general *k*-nn technique. Subsection 2.3.2 reviews the fuzzy *k*-nn technique. Subsection 2.3.4 describes special properties of the fuzzy *k*-nn technique.

2.3.1 *k*-nearest neighbors technique

The foundation of the fuzzy *k*-nn technique is the *k*-nn technique. The definition of *k*-nearest neighbors is trivial: For a particular point in question, in a population of points, the *k* points in the population that are nearest to the point in question. Finding the *k*-nearest neighbors reliably and efficiently can be difficult. How "nearness" is best measured and how data is best organized are challenging, non-trivial problems.

The implicit assumption in using any *k*-nearest neighbors technique is that items with similar attributes tend to cluster together. Hence, if unknown attributes of an observed item are sought, then examination of its neighbors should suggest what those unknown attributes may be. Any *k*-nearest neighbors technique is effective only to the extent that the assumption of clustering behavior is correct. But clustering behavior varies. For example, wealthy suburb dwellers tend to have wealthy neighbors, whereas wealthy city dwellers tend to have mixed "classes" of neighbors. The *k*-nearest neighbors method is most frequently used to tentatively classify points when firm class bounds are not established.

The vital part of any nearest neighbors technique is an "appropriate" distance metric, or similarity-measuring function, as Dudani (1976) explains.

> It is reasonable to assume that observations which are close together (according to some appropriate metric) will have the same classification. Furthermore, it is also reasonable to say that one might wish to weight the evidence of a neighbor close to an unclassified observation more heavily than the weight of another neighbor which is at a greater distance from the unclassified observation. Therefore, one would like to have a weighting function which varies with the distance between the sample and the considered neighbor in such a manner that the value decreases with increasing sample-to-neighbor distance.

Dudani (1976) proceeds to describe a "distance-weighted *k*-nearest neighbors rule" which can decide classification. Suppose we wish to label a new unknown point. We can decide its label according to the labels of its neighbors, either according to a distance-weighted scheme or according to a simple majority of its neighbors. In an experiment, Dudani (1976) found that "a lower probability of misclassification was obtained for the distance-weighted *k*-nearest neighbors rule than for a simple majority *k*-nearest neighbors rule."

The main problem to solve in developing any *k*-nn technique is to develop an *appropriate metric*. This problem occurs in numerous AI settings. Saying that a *k*-nn technique is "distance-based" leads to two questions:

- How is distance determined?
- How is uncertainty computed?

Kamp et al. (1998) explain how CBR, information retrieval systems, and database management systems (DBMS) all involve problem of finding nearest neighbors and of computing with uncertainty as follows.

> Handling of incomplete and vague data is closely intertwined with case-based reasoning. However, also within database research there is a growing interest in handling of incomplete and vague data. [In all these settings] nearest-neighbor queries can be directly used to determine the most similar cases.

Kamp et al. (1998) consider multidimensional access methods and contrast three query types: exact match, range, and nearest-neighbor. They offer a general warning about spatial access methods:

> It is often argued, or implicitly assumed that cases are points in <u>*d*</u>-dimensional space and the retrieval process is built upon this assumption. In our experience, this assumption is wrong. Most often, cases are incompletely described; the values of certain attributes are unknown, or only vague.

Kamp et al. (1998) suggest that the greatest opportunity for the development of CBR systems is scaling up systems and integrating them with existing large databases. Kamp et al. (1998) further suggest that

> a topic of future research within intelligent retrieval is the integration of domain knowledge and background knowledge to enhance the semantic of the retrieval. This could be done by considering and integrating techniques from knowledge representation [and] in this area, further research includes finding guidelines for finding the right tradeoff between expressiveness and complexity for different application scenarios, the search for approximations, etc.

Kamp et al. (1998) wrote the summary chapter in a book that compiles recent work by CBR researchers in Germany. The need for better schemes for approximation and dealing with imprecision is mentioned throughout the book. The word *fuzzy* is strangely absent from this book. Computing with imprecision is what fuzzy logic excels at.

> [fuzzy logic has a] *tolerance for imprecision* which can be exploited to achieve tractability, robustness, low solution cost, and better rapport with reality. (Zadeh 1999).

The fuzzy *k*-nearest neighbors technique described in the following section is a method that combines the advantageous distance-weighted quality recommended by Dudani (1976) with the need for better retrieval semantics (i.e., deal with vague attributes, incorporate domain knowledge, balance expressivity and complexity) recommended by Kamp et al. (1998).

2.3.2 Fuzzy *k*-nearest neighbors technique

A fuzzy k-nearest neighbors (fuzzy *k*-nn) technique is simply a nearest neighbors technique in which the basic measurement technique is fuzzy.

As detailed in the previous section, the term *nearest neighbors technique* refers to a technique that identifies the closest points to a given point in some multi-dimensional space. The motivation for using nearest neighbors techniques is usually to infer what one of the properties of an item is by examining other properties. The underlying assumption is that similar items cluster.

For a simple example, with a given a weather situation, suppose we want to determine what the precipitation type is, rain or snow. And suppose all we know is that the temperature is -10ºC and that it is precipitating. Examination of all such weather situations would show that precipitation types were usually snow, rarely some other form of frozen or freezing precipitation, and never rain. So, on that basis, we could infer that the precipitation type for the given situation is probably snow. We could classify the precipitation type as “probably snow.” Indeed, nearest neighbors techniques are usually used to enable some form of classification. In this thesis, we are more concerned with case-to-case comparison than with case-to-class comparison.

The fuzzy *k*-nn technique applied to continuous vectors, achieves automatic, expert-like similarity comparison of complex objects. Each comparable attribute in two comparable vectors is compared with an attribute-specific fuzzy set and the results are aggregated to describe the overall similarity of the two vectors. The fuzzy *k*-nn technique, applied to time series of continuous vectors, achieves effective analog prediction, as we demonstrated by developing a weather prediction system (Hansen and Riordan 1998). [46]

Keller et al. (1985) define and describe the fuzzy *k*-nearest neighbors algorithm in a foundational and theoretical paper. The key points of that article are summarized as follows.

Stored cases are labeled into distinct classes. Given a new case to classify, both crisp and fuzzy *k*-nn algorithms are made to find *k*-nn. The algorithms differ in two ways. First, the crisp algorithm uses a distance function, whereas the fuzzy algorithm uses fuzzy set based comparisons. Second, the crisp algorithm assigns the new case to the class that is represented by a majority of its *k*-nn, whereas the fuzzy algorithm assigns the new case varying degrees of membership to all the classes represented by the *k*-nn, according to the degree to which the new case matches each of the *k*-nn.

The fuzzy *k*-nn algorithm determines the degree of membership of any given continuous vector in any class of continuous vectors, as Keller et al. (1985) explain:

> The advantage provided by fuzzy sets is that the degree of membership in a set can be specified, rather than just the binary is or isn't a member. This can be especially advantageous in pattern recognition.

Patterns in real-world data are often ambiguous and therefore difficult to classify into crisp sets.

The fuzzy *k*-nn algorithm described by Keller et al. (1985) is general. It can be used to classify any kinds of continuous vectors into arbitrary classes. In this thesis, continuous vectors describe time-varying weather information. Each new, unique, present case weather acts as a special class—the "class" which best describes present weather and which is centered on the present case. We want to find in the database *k* nearest neighbors for such a special class of interest.

---

[46] To compare vector time series, we used matrix representations: rows for attributes, columns for time steps. (Hansen and Riordan 1998)

Comparing the fuzzy *k*-nn algorithm to a "crisp" *k*-nn algorithm, Keller et al. (1985) point out a special advantages of the fuzzy *k*-nn algorithm:

> An incorrectly classified sample will not have a membership in any class close to one while a correctly classified sample does posses a membership in the correct class close to one.

Keller et al. (1985) compared the performance of the fuzzy *k*-nn nearest neighbors with crisp *k*-nn algorithm (*k*-means clustering) and found two advantages in fuzzy *k*-nn method:

- Fuzzy *k*-nn classification is more accurate than crisp *k*-nn classification. [47]
- Fuzzy *k*-nn classification solutions include useful confidence measures based on to the resulting memberships.

Two contributions of fuzzy logic to case-based reasoning are that it can *improve performance of retrieval in terms of accuracy* and that it can *increase the interpretability of results of retrieval*. The accuracy improvement comes from the avoidance of unrealistic absolute classification. The interpretability increase comes from the overall degree of membership of a case in a class which provides a level of assurance to accompany the classification. For example (and to preview the way we use fuzzy logic in Chapters 3 and 4), if an expert user configures similarity-measuring fuzzy sets, and the thus-equipped similarity-measuring algorithm uses a simple *maximum* operation to aggregate results of numerous similarity tests, and the algorithm reports that a past case has an overall score of "very similar" to a present case, then the user can be assured that every attribute of the past case is at least "very similar" to the present case in accordance with that user's own expressed understanding of similar.

---

[47] The fuzzy *k*-nn algorithm achieves improved accuracy through the avoidance of unrealistic absolute classification during algorithm execution. Keller et al. (1985) claim, "The advantage is that no arbitrary assignments are made by the algorithm." By "not arbitrary," they mean: "The advantage provided by the fuzzy sets is that the degree of membership in a set can be specified, rather than just the binary is or isn't a member."

The algorithm for a classic fuzzy nearest neighbors prototype algorithm, proposed by Keller et al. (1985), is shown in Figure 8.

Let $W = \{Z_1, Z_2, \ldots, Z_c\}$ be the set of c prototypes representing c classes.

BEGIN
  Input x, vector to be classified.
  Initialize $i = 1$.
  DO UNTIL (distance from each prototype to x is computed)
    Compute distance from Z_i to x.
    Increment i.
  END DO UNTIL
  Initialize $i =1$.
  DO UNTIL (x assigned membership in all classes)
    Compute $u_i(x)$ using (1)
    Increment i.
  END DO UNTIL
END

where

$$u_i(x) = \frac{1 / \|x - Z_i\|^{2/(m-1)}}{\sum_{j=1}^{c} (1 / \|x - Z_j\|^{2/(m-1)})} \qquad (1)$$

where m determines how heavily the distance is weighted when calculating each neighbor's contribution to the membership value.

and where $\|x - Z_i\|$ represents the membership of x in the class of Z_i

Figure 8. Fuzzy nearest prototype algorithm copied from (Keller et al. 1985).

An item may be defined by a collection of continuous attributes and is thus general. A class may be defined by either a specific case (a prototype), an idealized case, or a group of labeled items, and is thus general. The assumption in using prototypes is that prototypes are complete members of the class that they represent.

In addition to comparing fuzzy and crisp k-nn algorithms, Keller et al. (1985) compare fuzzy and crisp nearest prototype classifiers:

> Actually, the only difference is that for the nearest prototype classifier, the labeled samples are a set of class prototypes, whereas in the nearest neighbors classifier we use a set of labeled samples that are not necessarily prototypical. Of course, the nearest prototype classifier could be extended to multiple prototypes representing each class, similar to the k-nearest neighbors routine.

To classify a new case using the nearest prototype classifier,

> membership in each class is assigned based only on the distance from the prototype(s) of the class.

Keller et al. (1985) describe the benefits of the fuzzy prototype algorithm:

> The fuzzy prototype classifier, while not producing error rates as low as the fuzzy nearest neighbors classifier, is computationally attractive and also produces membership assignments that are desirable.

The membership provides a useful level of confidence of the classification.

Our fuzzy *k*-nn algorithm, described in Chapter 3, may be viewed as a variation of the above algorithm, a variation that includes some steps to reduce computational cost, as explained in Section 3.3.1 (page 89).

2.3.3 Weather situations are not prototypical

Weather situations cannot be condensed and faithfully represented as distinct prototypes. There are no original models of weather situations upon which all other weather situations are patterned: each situation is unique (as explained in section 1.5.3, page 26).

Therefore, to achieve the benefits of the fuzzy nearest prototype algorithm for analog weather prediction, we will treat the present weather case as the only “prototype” against which all stored cases must be compared. Given a new, unique weather case, and a database of past weather cases, we wish to isolate *k* analogs for the present case among the past cases. The challenging problem is to determine the degree to which past cases are analogs. The subsequent problems of sorting, weighting, and predicting are relatively simple.

Two basic contrasting approaches are crisp *k*-nn and fuzzy *k*-nn. With the crisp *k*-nn one would in advance specify thresholds (ranges) for membership in the set of analogs. If past cases fall within the ranges they would be classified as analogs, and if past cases fall outside the ranges they would be classified as non-analogs. A problem with this approach is that it is unlikely that it will isolate *k* analogs—it may produce no matches, fewer than *k* matches, or more than *k* matches—because some weather cases are rare while others are common. This is the point at which arbitrary search parameter adjustments may need to be made by a user, and thus algorithm autonomy is lost. To isolate *k* analogs certain ranges would need to be widened or narrowed. This would require user intervention and perhaps successive attempts. Each such range-adjusting intervention is arbitrary, for instance, should one tighten the “temperature fit” or the “wind direction fit.” Empty prescribed bins have been a problem in earlier comparable weather analog forecasting systems (Martin 1972).

The fuzzy *k*-nn algorithm is based on the ideas that analogs are similar cases and that *similar* is a fuzzy property. For example, if the present temperature is 10°C, then a temperature of 11°C would be considered very similar, 15°C somewhat similar, and 20°C hardly similar.

The fuzzy *k*-nn algorithm can run autonomously because no arbitrary classification assignments are made by the algorithm. Rather, the fuzzy *k*-nn algorithm describes all past cases as having varying degrees of membership in the set of analogs for the present case, sorts them, and the *k* cases with the highest degrees of membership are the *k*-nn. With a fuzzy distance metric, one pass through the set of past cases will always isolate *k*-nn. Fuzzy sets emulate arbitrator experts. For example, for purposes of evaluating similarity, a weather forecasting expert may consider a difference of 5 degrees Celsius in temperature as equivalent to a difference of 10 degrees in wind direction. For case-to-case comparison, triangular fuzzy sets are specified accordingly.

The degree to which the fuzzy *k*-nn are analogs for the present case make the search results interpretable. If the *k*-nn are analogs to a high degree, then the current case is common and one can associate high confidence in a prediction based on the *k*-nn. If the *k*-nn are analogs to a low degree, then the current case is rare and one can associate low confidence in a prediction based on the *k*-nn.

2.3.4 Properties of the fuzzy *k*-nn technique

In subsection 2.3.1, we explained how three aims of research into intelligent retrieval are: to integrate domain knowledge into the retrieval process, to deal sensibly with the uncertainty associated with approximately measured variables, and to find the right balance between expressivity and complexity (Kamp et al. 1998). In subsection 2.3.2, we explained how fuzzy set theory is a framework that combines realistic expressions of domain knowledge with uncertain variables, controls expressivity through the shape of fuzzy sets, and controls complexity through the aggregation of a collection of system-modelling fuzzy statements. In this subsection, we describe three special properties of the fuzzy *k*-nn technique that help it to perform intelligent retrieval.

First, the fuzzy *k*-nn technique recognizes family resemblance. Pattern recognition depends on being able to detect and describe similarity between objects. Dreyfus (1992) describes two approaches for describing similarity between objects, family resemblance and class membership, as follows.

> Family resemblance differs from class membership in several important ways: classes can be defined in terms of traits even if they have no members, whereas family resemblances are recognized only in terms of real of or imaginary examples. Moreover, whereas class membership is all or nothing, family resemblance allows a spectrum ranging from the typical to the atypical.

The fuzzy *k*-nn technique does not define classes in terms of specific traits. It does not determine degrees of membership of a weather case in predefined fuzzy sets such as *low* or *high* temperature. Rather, it measures the degree to which any given case is *similar* to other cases in the database. In the sense of (Dreyfus 1992), it determines the degree of family resemblances of the case compared to the *k* most similar cases in the entire population of cases.

Second, the fuzzy *k*-nn technique avoids the cluster validity problem—because it is not a "fuzzy clustering technique." The purpose of clustering techniques is to identify structure, or to recognize patterns. [48] Whereas the purpose of fuzzy *k*-nn is simply to identify similarity, not to delineate presumed clusters.

Zimmerman (1991) describes fuzzy clustering and we summarize his description as follows: One begins by assuming that the problem of feature extraction has been solved. Each of n items is characterized by p attributes and the task is to divide the items into c categories, $2 \leq c < n$, homogeneous subsets called "clusters." The number of clusters, c, is usually not known in advance.

Computation of fuzzy *k*-nn is a preliminary step in the computation of fuzzy clusters. A point with more near neighbors than any other point is, logically, the center of a cluster. Zimmerman (1991) describes how the centers and memberships are calculated by using an iterative scheme to minimize a summation of matrices. The details of this scheme are not relevant here, so we will forgo them. What is relevant here, about fuzzy clustering, is the question of *cluster validity*.

Yang (1993) reviewed 103 papers dealing with fuzzy clustering and concluded:

> We have already reviewed numerous fuzzy clustering algorithms. But it is necessary to presume the number c of clusters for all these algorithms. In general, the number c should be unknown. Therefore the method to find optimal c is very important. This kind of problem is called cluster validity.

---

[48] Bezdek and Pal (1992) offer a large collection of papers which describe how to use fuzzy models for pattern recognition.

The fuzzy *k*-nn avoids the difficult cluster validity problem by forgoing the problem of identifying global patterns—it assumes that such patterns exist and forgoes the problem of describing such patterns—and, instead, for a specific case proceeds to identify the most relevant information, local patterns, contained in a case base by identifying *k*-nn for that case.

Third, the fuzzy *k*-nn technique avoids the "rule explosion problem"—because it is not a conventional fuzzy rule based system. Rule explosion is a major problem in designing fuzzy rule based systems. The number of rules in a system grows exponentially with the number of input and output variables (Kosko 1997). [49] For instance, suppose we have a system where each input is represented by three fuzzy sets. If we have two inputs, then nine fuzzy rules are needed ($3^2=9$). If we have three inputs then twenty-seven fuzzy rules are needed ($3^3=27$). Hence, fuzzy systems do not scale up well for systems with many variables, or dimensions.

The fuzzy *k*-nn algorithm avoids the rule explosion problem because it does not attempt to abstract the complex behavior of a system into rules. Instead, each observed system state is, in effect, a rule and, in that sense, the ratio of rules to model points is contained at 1:1. A fuzzy rule and a fuzzy similarity-measuring function are contrasted in Figure 9.

$$a_1 \vee b_1 \rightarrow c_{11}$$
$$\ldots$$
$$a_i \vee b_j \rightarrow c_{ij}$$

$$sim\,((a_1, b_1), (a_2, b_2)) = \mu_a(a_1, a_2) \vee \mu_b(b_1, b_2)$$

(a) Fuzzy rule based system. Input variables are *a* and *b*, and the output variable is *c*. Input variable *a* is represented with *i* fuzzy sets and variable *b* is represented with *j* fuzzy sets. The number of required rules equals the product of $i \cdot j$. As input variables are added, the number of rules grows exponentially.

(b) Fuzzy similarity-measuring function. (a_1, b_1) and (a_2, b_2) describe two states of a 2-D system. μ_a and μ_b are two similarity-measuring functions used to compare two attributes. The number of required similarity-measuring operations rules equals the number of variables of the system. As cases are accumulated, the number of operations grows linearly.

Figure 9. Rule explosion in a fuzzy rule base contrasts with rule containment in a fuzzy *k*-nn similarity-measuring function

---

[49] In addition to an exponential increase in number of rules as the numbers of input and output increase, there is a linear increase in number of rules as the expressed precision of input and output dimensions

A possible contribution of case-based reasoning to the field of fuzzy logic based applications would be to help to avoid the rule explosion problem. For the fuzzy *k*-nn technique, the number of necessary similarity-measuring functions grows linearly with the number of input dimensions, not exponentially.

2.4 Applications that use fuzzy *k*-nn techniques

This section surveys numerous applications that exemplify fuzzy *k*-nn techniques. What all these applications have in common is that, in one way or another, they all perform fuzzy logic based matching and exploit descriptions of salient attributes and combinations of attributes acquired from domain experts using fuzzy words. Fuzzy logic directly translates expert descriptions of salient features into a fuzzy set based similarity-measuring algorithms.

Applications surveyed include weather prediction, mergers and acquisitions, residential property valuation, cash flow forecasting, shoe fashion database retrieval, colour matching in plastics production, criminal profiling, identifying freshwater invertebrates, interpreting electronic nose data, and manufacturing failure analysis.

2.4.1 Weather prediction

We built a fuzzy *k*-nearest neighbors based weather prediction system (Hansen and Riordan 1998). The fuzzy *k*-nn method is used to acquire knowledge about what salient features of continuous-vector, unique temporal cases indicate significant similarity between cases. Such knowledge is encoded in a similarity-measuring function and thereby used to retrieve *k* nearest neighbors from a large database. Predictions for the present case are made from a weighted median of the outcomes of analogous past cases, the *k*-nn. Past cases are weighted according to their degree of similarity to the present case.

Numerically described attributes are fuzzified into memberships in specific fuzzy sets before being compared. The fuzzy *k*-nn system fits fuzzy sets to the attributes. Because we search for nearest neighbors for a particular case, the centers of triangular fuzzy sets are centered on the attribute values of that particular case.

---

increases. For example, an input may be represented by two fuzzy sets (*low* and *high*) or more precisely by thee fuzzy sets (*low*, *medium*, and *high*).

2.4.2 Mergers and acquisitions

Bonissone and Ayub (1992) developed a system to predict the outcome of mergers and acquisitions. The system could give guidance about what companies might be worth acquiring. The case base has descriptions of companies where attributes describe various economic conditions. Descriptions of past cases refer to five temporal phases: initial conditions, pre-tender, tender-negotiation, outcome, and long-term results. Descriptions of new cases (i.e., prospective purchases) refer to only initial conditions. The system determines the degree to which new cases match past good purchases and bad purchases.

The motivation for applying CBR, as is typical, is to cope with a domain where the domain theory is weak and where knowledge acquisition is difficult.

The motivation for applying fuzzy methods to CBR is to deal with *uncertainty* in four phases of the CBR process: semantics of abstract features used to describe cases, evaluation of the similarity measures computed across these features, determination of relevancy and saliency of the similar cases, and the solution-adaptation phase.

The system is divided into two parts, one dealing with domain knowledge, the other dealing with case representation. Domain knowledge is organized into three hierarchies: objects, action, and goals.

Individual abstract features are described by applying domain rules to numerous surface features. This object-oriented strategy of bundling related attributes together reduces the dimensionality and thus the complexity of the problem. [50]

2.4.3 Residential property valuation

Bonissone and Cheetham (1997) developed a system to valuate residential properties. The case base has descriptions of houses in mortgage packages and descriptions of current market conditions. The case base describes hundreds of thousands of real estate transactions. Mortgage packages are investment tools that may contain up to 1000 mortgages. Real estate markets are volatile. A valuation system for mortgages helps an investor to keep better track of their real estate investment.

Previous efforts to automate valuation failed to capture the intrinsic imprecision in sale comparison. Imprecision surrounds terms in the problem: “find the *most similar* houses, located

---

[50] We used such an object-oriented strategy in (Hansen and Riordan 1998) and use it again in this thesis. For example, wind is an abstract attribute of a weather case that is composed of surface attributes such as wind speed, wind direction, and "wind run."

close to the subject property, sold *not too long* ago; and selecting a *balanced* subset of the most promising comparables to derive the final estimate."

Fuzzy sets to measure these properties were constructed by interviewing experts, asking them to describe their preferences for various properties, and making fuzzy sets accordingly. The system was designed to permit the similarity-measuring fuzzy sets to relax (i.e., widen) if the retrieved set was too small.

Our fuzzy *k*-nn approach, using triangular sets which taper off asymptotically, avoids the problem of retrieving too small a set. The efficacy of this approach, compared to using support-limited trapezoidal sets, is also implied by the conclusion of Liao et al. (1998): "the more the fuzziness in the case attributes, the more the power of the [fuzzy similarity] measure."

The fuzzy CBR system was tested against three other methods: a statistical formula, a fuzzy-neural net, and a human appraiser. The fuzzy CBR system was more accurate than the first two objective methods and a bit less accurate than a human appraiser.

Bonissone and Cheetham (1997) emphasize the efficacy of the system at producing confidence value assessments to accompany house value estimates. The distribution of the retrieved analogs itself implies how much confidence to place in an estimate.

Bonissone and Cheetham (1997) emphasize the *transparency* of the system workings. Every step used to determine a house value can be reviewed and readily understood by users. The decisions and weights correspond to intuitively understandable expressions that coincide with the specifications of the users, the interviewed experts. Thus fuzzy CBR achieves a unique form of explanation capability.

2.4.4 Cash flow forecasting

Weber-Lee et al. (1995) combine fuzzy logic and CBR in a cash flow prediction system. The most original aspect of their system, as they point out, is the use of a fuzzy Sugeno integral to calculate the similarity of two financial situations rather than the usual weighted mean approach. [51] This integral calculates the overall similarity according to the "max of the min" of the individual attribute to attribute similarities.

The "max-min" scheme is the simplest possible aggregation scheme for additive fuzzy systems, and it has the mathematical properties of associatively, reflexivity, symmetry,

[51] Note that Weber-Lee et al. (1995) refer to the *weighted mean approach* as it is used for individual similar case recognition, not for subsequent solution composition. In our system, we will use a sort of weighted mean to compose solutions base on an analog ensemble of weather cases.

transitivity (Zimmerman 1991). There are more complicated and sophisticated aggregation schemes, for dealing with additive systems and fuzzy rule bases, [52] but Weber-Lee et al. (1995) use the simplifying assumption that overall similarity is as only strong as the weakest individual similarity and their encouraging results support this assumption. We regard weather the same way when we apply the fuzzy *k*-nn method.

Weber-Lee et al. (1995) make an appealing argument for using CBR to forecast cash flow, an argument which could easily be adapted to the problem for forecasting weather, as follows.

> The task of forecasting cash flows when performed by a human being works adequately under similar and sequential contexts. The expert aggregates information [such] as a possible recession and becomes subjectively pessimistic.. After a while, the expert cannot remember if the pessimistic approach used, for instance, 5 years ago, actually turned out to be effective, and if so, how effective. The CBR approach overcomes this shortcoming.

One of the main attributes used by Weber-Lee et al. (1995) for CBR-based prediction is that of season itself. Weather patterns change continually as seasons change, and memories of weather situations fade over time. Weather analogs from similar dates in previous years are better analogs than analogs from more recent dates in previous months. Analog forecasting preserves relevant memories of analogous situations from previous years.

2.4.5 Shoe fashion database retrieval

In an application for the shoe fashion industry, Main et al. (1996) used fuzzy feature vectors to enable retrieval of patterns for shoes similar to actual or idealized shoes. The motive was to rapidly manufacture "new" styles of shoes, to imitate currently popular models, by recycling old similar styles. After testing the fuzzy features approach for case selection, they found that "the cases retrieved matched the current case the closest in at least 95% of the tests." The fuzzy vector based retrieval agreed with experts' judgements of what constituted a *high, medium*, or *low* dimension on various parts of shoes

2.4.6 Colour matching in plastics production

Cheetham and Graf (1997) built a system to perform color matching in plastic manufacturing. The goal was to determine an optimal combination of colorants to create a

---

[52] The Standard Additive Model (SAM) or Center of Gravity (COG) are commonly used fuzzy aggregation schemes (Kosko 1997), schemes used to fuzzify and defuzzify system throughput. COG and SAM are commonly used to compute the output of fuzzy rule bases consisting of 1000's of rules. Figure 9 on page 64 compares the SAM approach and the "max-of-the-min" approach.

specific color of plastic and to do so at minimal cost. The case base consisted of past "recipes" and results. The new case consisted of a color sample to be matched. The system was used for two years at a number of General Electric Plastics sites and lead to significant cost savings.

Fuzzy logic was used to measure the level of satisfaction with several diverse factors, such as match under different lighting conditions, cost of colorants, and opacity of resultant plastic.

The most important concept in the system is that of a *fuzzy preference function*. Cases are compared and the differences are recorded with real numbers. An expert specifies thresholds that correspond to linguistic terms describing the quality of match. Fuzzy sets for attribute comparisons are constructed accordingly. Five such fuzzy sets enable the simultaneous comparison of five heterogeneous attributes.

> Each of the [five properties] is based on different scales of units. By mapping each of these properties to a global scale through the use of fuzzy preferences and linguistic terms such as Excellent, Good, Fair and Poor, it becomes possible to compare one attribute with another.

This integrated approach to comparison of diverse attributes is similar to the fuzzy *k*-nn weather prediction system. For example, 10 degrees difference is *near* for wind direction, 5 degrees difference is *near* for temperature, and so on.

2.4.7 Criminal profiling

In an application to profile criminals, Lefley and Austin (1997) used fuzzy methods to describe criminals *modus operendi* (MO). An MO is a learned pattern of criminal behavior which a particular habitual criminal tends to follow and which detectives use to identify the particular criminal. Such behaviors are weakly indicative individually and strongly indicative collectively. Unsolved crimes may be solved by associating known offenders with the crimes and investigating those offenders more closely.

Their algorithm used a fuzzy distance measure of several attributes, forward-chaining logic, and calculated similarity according to a summation of individual attribute results.

In practice, the application was simple. Records of criminals and crimes were obtained and several crimes were presented as "unsolved" (i.e., the criminals were not identified). Students reviewed the records all the records and transferred what they read into a questionnaire, a form which the system could process directly. The values in the form were fuzzy descriptors of criminal attributes. In experiments, they found that most students transferred common written criminal records into the fuzzy forms consistently.

Their system performance was encouraging. "Unsolved" crimes were measured as most similar to solved crimes of the actual offender in 70% of the trials. Their conclusion emphasizes the ease, effectiveness and envisioned portability of the matching system.

The system exploits available expertise. The system questionnaires were designed based on important attributes suggested by criminologists.

2.4.8 Identifying freshwater invertebrates

Winder et al. (1997) identify freshwater invertebrates using an approach which is very similar to that used by Lefley and Austin (1997) to identify criminals. Their system identifies samples of invertebrates based on questions answered using a database of characteristics of species suggested by taxonomists.

Again, knowledge is acquired from experts about salient attributes with which to identify individuals, in this case species rather than criminals. Thereafter system construction and use is straightforward. Construction requires converting the knowledge of salient attributes into fuzzy aggregate matching operations. To use the system, the user fills in a questionnaire.

Winder et al. (1997) also note how their fuzzy CBR approach to measuring similarity aims to associate new cases with *similar families*. As we noted above (subsection 2.3.4, page 62), fuzzy *k*-nn is a useful tool for the problem of *recognizing family resemblance* between items, a challenge for AI described by Dreyfus (1992).

2.4.9 Interpreting electronic nose data

Singh (1999) explains how to reduce the ambiguity of classification of a new point by centering a fuzzy *k*-nn algorithm on the new point. Using the *single* fuzzy nearest-neighbor is most effective for two pattern recognition problems, a benchmark problem and a realistic problem. [53] The algorithm labels a new case according to the label of its single nearest neighbor, as measured with fuzzy operations, rather than according to the majority of the labels of its *k* nearest neighbors (*k*-nn). The crisp *k*-nn approach they refer to, basically, draws a hypersphere around the point to be labeled, counts the various labels of points within the hypersphere, and assigns to the point to be labeled the label which describes the majority of the points in the

---

[53] The two pattern recognition problems attempted by Singh (1999) are detection of spirals in a commonly attempted benchmark problem, and the realistic problem of identification of different blends of coffee according to data collected from an "electronic nose." Singh (1999) does not refer to (Keller et al. 1985) but the basic approach of the fuzzy k-nn algorithm is the same. Singh (1999) does not deal with temporal aspects of the spiral data.

hypersphere. There are three problems with this crisp *k*-nn approach: it results in confusion if there is no clear majority in the hypersphere, results are sensitive to the radius of the hypersphere, and it works poorly for nonlinear data where clusters of different types often overlap near the point to be labeled. The fuzzy single nearest neighbor approach is better, Singh (1999) claims, because it detects the "truly nearest neighbor." [54]

2.4.10 Electronics manufacturing diagnosis

Göös et al. (1999) describe a fuzzy monitoring and case-based diagnosis tool for an electronics manufacturing process. The tool provides online quality control during manufacturing of electronics parts. Components of parts are measured during manufacturing. Each part (case) is described by 19 attributes, most of these being measured attributes. Three values are associated with each attribute: a lower threshold for acceptance, and optimal value, and an upper threshold for acceptance. The quality of a part is determined according to the degree to which its attributes collectively are similar or dissimilar to the theoretically optimal part.

The case base consists of descriptions of past low quality parts along with descriptions of the cause and remedy of each such situation. When a new part is diagnosed as of low quality, a case base search is performed. A nearest neighbors algorithm with expert-specified weights is used to determine overall quality of the part. Continuous measurements map to linguistic (fuzzy) descriptions of quality.

Case base searches lead to one of two sorts of useful results. First, either a near neighbor is found and the cause of the defect is generally diagnosed and a remedy found. Or, second, a new sort of defect is identified, the user is alerted to this anomaly, and the user revises the case base accordingly.

Results and user feedback from initial fielding of the system are quite positive (Göös et al. 1999). The method is effective at finding manufacturing defects. Users readily accept the results, appreciate the opportunity to learn from the results, and appreciate the opportunity to, in

[54] Singh (1999) argues, somewhat rhetorically, and with an idealized diagram, for the superiority of a single fuzzy nearest neighbor approach over a non-fuzzy k-nn approach, but tests the single fuzzy nearest neighbor approach against a neural network. The results are encouraging, but still leave us to wonder, if considering only nearest neighbor approaches, what the effects are of reducing *k*—to as low as *k*=1? (Singh (1999) intuitively selects *k*=1.) and what are the effects using of fuzzy measures versus crisp measures. We will examine the effects of varying *k* and of varying fuzziness in our experiments (Section 4.2, page 102).

effect, teach the system when anomalous new defects are detected and the case base needs to be revised.

Tobin et al. (1998) use a fuzzy *k*-nn classification algorithm to enable a system for semiconductor defect detection. Their algorithm follows closely from the fuzzy *k*-nn algorithm formulation of Keller et al. (1985). The training set of cases consists of records of defective semiconductors. Each record has 35 measured features (measured with optical instruments) and one of 14 types of classes. Classes describe the type of defect and the corrective measure, and are determined by experts. Their system showed a "similar efficacy to the human counterpart." It worked well in the assembly line environment, allowing for a automatic 100% inspection rate, thus the system was judged useful for quality control.

2.5 CBR and fuzzy logic based weather prediction systems

This following three subsections survey weather prediction systems that use CBR, fuzzy logic, and the combination of both. The section *Additional References on Analog Forecasting in Meteorology*, at the end of this thesis, lists meteorology articles that describe the challenges of implementing analog forecasting. We do not survey these articles here as they are too far from the thesis subjects of CBR and fuzzy logic.

2.5.1 CBR weather prediction systems

Jones and Roydhouse (1995) are among the few who have applied intelligent retrieval to archived meteorological data and referred to CBR. [55] Their motive is the same as ours: to exploit large meteorological archives. Jones and Roydhouse (1995) use a structured database approach for retrieving similar, large-scale weather maps (maps that contain abstract structures, such as a *high pressure area* or a *cold front*), whereas we use a relational database approach for retrieving similar, local-scale weather reports (reports that contain related variables, such as cloud ceiling height 200 feet and visibility in fog 0.5 miles).

Jones and Roydhouse (1995) describe a prototype system called *Metvuw* which is intended to help forecasters predict the evolution of cold fronts over the Tasman Sea, between Australia and New Zealand. Forecasting fronts in this area is difficult because there are few weather observations in that area, thus humans are poorly informed and computer models are

[55] Jones and Roydhouse (1995) cite two of their own previous related work. We have found no subsequent papers by these authors describing related work.

poorly initialized. They suggest that examination of the evolution of analogous previous weather situations may help forecasters predict for the present situation.

Forecasters provide Metvuw with high-level descriptions of current large-scale, atmospheric pressure patterns (high and low pressure centers and values) and Metvuw retrieves similar cases (i.e., relevant and analogous cases) from a database of 2500 stored cases. Jones and Roydhouse (1995) use a similarity-measuring algorithm which assesses various salient pressure features suggested by meteorologists. [56]

To minimize computational cost, Jones and Roydhouse (1995) use a two-stage screening process. The stage selects a few candidate cases using a set of rough tests. The second stage ranks the few selected cases using more a costly but accurate process of similarity assessment. Such a strategy is often referred to in CBR as "MAC/FAC," short for Many Are Collected, Few Are Chosen" (as described in subsection 2.1.1 page 42). We use such a strategy when we implement the fuzzy *k*-nn system (described in Chapter 3).

Although the Metvuw system showed some promise, its development apparently ceased five years ago. We can only speculate as to why, but perhaps one of the following qualities, which we will try to avoid, contributed to cessation of the system's development.

- *Non-autonomy.* "Metvuw Workbench assists the retrieval of relevant past cases, but leaves their adaptation and interpretation to the user." (Jones and Roydhouse 1995). Forecasters are already deluged with data to interpret as it is. As Conway (1989) points, systems must be convenient and fast for forecasters if they are to be useful. We envision fuzzy *k*-nn weather prediction as being able to operate in two modes: *autonomous* guidance-offering, and *user-driven* weather archive querying.
- *Data-intensive.* Jones and Roydhouse (1995) expected that a 10-year data set for Metvuw would require approximately one Gigabyte to store, whereas Hansen and Riordan (1998) found that a 36-year data set required 6 Megabytes to store. Metvuw requires roughly three orders of magnitude more data storage because it its cases consist of images describing a large part of the globe, whereas the cases in (Hansen and Riordan 1998) consist of condensed numerical weather observations for one point, an airport.

---

[56] Salient pressure features suggested by meteorologists include: locations of pressure mimima, aligned overlap of minima-bounding rectangles, aspect ratio, density, area, and intensity (Jones and Roydhouse 1995).

Bardossy et al. (1995) describe another algorithm that uses a fuzzy-rule base and which has basically the same purpose as that of (Jones and Roydhouse 1995), that is, classification of atmospheric circulation patterns. Their fuzzy rule based algorithm is better at pattern recognition than that of Metvuw. Bardossy et al. (1995) reliably determine the degrees of membership of each new pressure pattern in a set of prototypical templates of pressure patterns, whereas, Jones and Roydhouse (1995) represent all pressure patterns symbolically with trees and this causes misleading results (e.g., slight deepening of low pressure systems causes totally different tree representation to result). In terms of database strategy, Bardossy et al. (1995) use a more of a relational database strategy and Jones and Roydhouse (1995) use more of a structural database strategy.

2.5.2 Fuzzy weather prediction systems

Fuzzy logic has so far seldom been used to predict weather, even though such an application was proposed at least 19 years ago, [57] fuzzy logic itself was first described 35 years ago (Zadeh 1965), and fuzzy logic has during recent few years become a mainstream technique in a variety of environmental domains. It represents linguistically-expressed domain knowledge and operates on diverse forms of continuous data—such types of knowledge and data are typical in environmental problems. Environmental domains where fuzzy logic presently operates effectively include agriculture, climatology, earthquakes, ecology, fisheries, geography, geology, hydrology, meteorology, mining, natural resources, oceanography, petroleum industry, risk analysis, and waste management (Hansen et al. 1999).

---

[57] Silvert (1981) proposed the use of fuzzy logic to formulate and evaluate predictions and proposed its use for verification of weather predictions. [The same "novel" idea occurred to me 15 years later in 1996.] He identified four types of prediction: "*sharp* ('this horse will win the race'), *fuzzy* ('this horse will do well'), *conditional* ('this horse will do well if it doesn't rain'), and probabilistic ('odds on this horse are…')".

Note that the fuzzy type of prediction can, through design, emulate the other types of prediction. If "do well" equates to "winning," then the prediction is sharp. If additional conditions are added, then the prediction is conditional. If "do well" equates to "odds of winning over 90%," then the prediction is probabilistic.

We acknowledge the simmering (cooling?) debate between proponents of probability and proponents of fuzzy logic over the distinct merits of each methodology. However, we point out that probability and fuzzy logic are two distinct means to the same end—the formation of expectation—and thus complementary. Probability forms expectation based on frequencies of past events. Fuzzy logic forms expectation on based on vague rules; these rules can take into account past events, as we do in this thesis, and thus, fuzzy logic can be made to emulate probability. In any case, current indications are that both methodologies will persist, coexist, and provide complementary approaches for problem solving (Bezdek 1994).

Maner and Joyce (1997) built a weather prediction system, called WXSYS. They obtained simple weather prediction rules (i.e., weather lore) from experts and weather almanacs, and implemented these rules in a system using a fuzzy logic rule base. For example, one rule they used is: "Weather will be generally clear when the wind shifts to a westerly direction. The greatest change occurs when the wind shifts from east through south to west."

According to Maner and Joyce (1997), there are three reasons why fuzzy logic seems ideally suited for weather forecasting:

- The phrases used in conventional forecasts are inherently and intentionally fuzzy.
- "Fuzzy logic is known to work in this domain."
- "The weather domain meets the general conditions under which a fuzzy solution is thought to be appropriate."

Fuzzy logic has been used to build expert systems to predict fog and to predict wind. Sujitjorn et al. (1994) and Murtha (1995) separately built systems to predict fog at an airport. Hadjimichael et al. (1996) and Kuciauskas et al. (1998) together built a fuzzy system, called MEDEX, for forecasting gale force winds in the Mediterranean. All of these systems are conceptually based on the classic fuzzy rule base approach to fuzzy systems. [58] How they differ is in the particular fuzzy rules elicited from experts. For example, the MEDEX system uses rules of the form "if pressure gradient is very large…then…", and so on.

We built a fuzzy expert system for critiquing marine forecasts, called SIGMAR [59] (Hansen 1997). Like the above fuzzy expert systems, expert-specified fuzzy sets are at its core. Unlike the above fuzzy expert systems, it does not process a series of fuzzy rules (e.g., if *A* and *B* then *C*). Instead it measures similarity using fuzzy sets: it measures the similarity between a current valid marine forecast and the actual marine observations *directly* by using fuzzy sets, rather than, as is usually done, *indirectly* by using categories (e.g., "Observation in category 1 and forecast in category 2.").

Marine forecasters use observations to determine how the accuracy of a forecast is faring. If unforecast significant conditions develop, then forecasts must be amended as soon as possible. The task of monitoring weather observations and assessing their significance in terms

---

[58] The fuzzy rule base approach to expert systems is well explained by Zimmerman (1991). Kosko (1997) refers to the rule base as a "fuzzy associative memory" and describes the process of rule resolution as firing all rules partially and in parallel and take a balanced average. Viot (1993) describes a fuzzy rule based system balance an inverted pendulum (a benchmark problem for fuzzy systems) and convincingly demonstrates how simple the system is by providing compilable C code for the system on *one page*.
[59] SIGMAR is short for *Significant Information Generated from Marine Area Reports*.

of the current forecast is called “keeping a weather watch.” For weather forecasting operations, the task is necessary and important. For weather forecasters, the task of continuously monitoring a never-ending list of ever-changing numbers is challenging during periods of rapidly changing weather and boring during periods of slowly changing weather.

SIGMAR continuously critiques marine forecasts: it automatically monitors a stream of real-time of observations, assesses where and to what degree a forecast is accurate or inaccurate, and reports to forecasters. SIGMAR helps marine forecasters to quickly identify any wind reports that contradict the marine forecast. This helps forecasters to respond quickly in situations where marine forecasts need to be amended.

Actually, fuzzy expert systems and CBR systems for weather prediction overlap. Tag et al. (1996), following the example of Bardossy et al. (1995), used fuzzy logic to automate the recognition of patterns of upper air wind flow. This pattern information was used, in a CBR-like way, as predictive input in a fuzzy expert system (MEDEX, described in the previous subsection).

Clustering techniques can be useful for CBR. Fuzzy clustering has been used to emulate human expert classification of climate (McBratney and Moore 1985) and climatological circulation patterns (Bardossy et al. 1995).

To the best of our knowledge, our current line of work is the only work which combines the three topics of fuzzy logic, CBR and weather prediction in a single system (Hansen and Riordan 1998). Given a present incomplete weather case to predict for, we used a fuzzy *k*-nn algorithm to find similar past weather cases in a huge weather archive to make predictions from.

Granted, the individual three methods are, by themselves, basic: using fuzzy sets to measure similarity is a basic application of fuzzy set theory, *k*-nearest neighbors is a basic CBR method, and analog forecasting is a primitive weather prediction technique. But when these three methods were combined into one system, with an expert’s knowledge of what features are salient and how, with the knowledge encoded as fuzzy sets, and the system was provided with a huge archive of weather observations, the results were encouraging. The system’s prediction accuracy was measured with standard meteorological statistics and compared to a benchmark prediction technique, persistence. In realistic simulations, the system was significantly more accurate. This following three chapters build upon the work begun in (Hansen and Riordan 1998).

3. System for Fuzzy *k*-Nearest Neighbors Based Weather Prediction

In Chapter 1, we explained how retrieval of similar cases relates to CBR, fuzzy logic, and weather prediction—CBR depends on retrieval of similar cases, fuzzy logic enables retrieval of similar cases, and the weather prediction technique of analog forecasting depends on retrieval of similar cases—and explained how case adaptation and case authoring are regarded as main challenges in CBR system development. In Chapter 2, we described how, to overcome the case adaptation and case authoring problems of CBR system development, three strategies have been proposed—knowledge acquisition, partial matching, and use of very large cases bases—and surveyed the literature to provide a basis for using the fuzzy *k*-nn technique to combine these strategies into one technique.

In this chapter, we describe the WIND-1 system. The system uses a fuzzy *k*-nn algorithm designed to emulate a weather forecasting expert applying the weather prediction technique of analog forecasting and, thereby, shows how fuzzy logic based methods enable a CBR system developer to impart the perceptiveness and case-discriminating ability of a domain expert to CBR. We describe the necessary details in as general terms as general as possible. Though the WIND-1 system is specifically designed for weather prediction, similar systems should be applicable to other kinds of analog prediction problems where large databases exist and perceptive experts are available to help to tune similarity-measuring fuzzy sets.

The WIND-1 system consists of two main parts:

- *A large database of weather observations.* The database is a weather archive of over 300,000 consecutive hourly weather observations.
- *A fuzzy k-nn algorithm.* The algorithm measures the similarity between temporal cases, past and present intervals of weather observations. The algorithm is tuned with the help of a domain expert, in our case, a weather forecaster experienced in noting similarities between cases.

Section 3.1 describes the database: an archive of airport weather observations. Section 3.2 describes our fuzzy *k*-nn algorithm which searches the database for analogs. Section 3.3 compares our fuzzy *k*-nn approach case-based reasoning with two previous approaches. In the next Chapter 4, we will present a set of experiments and results based upon the WIND-1 system.

3.1 Large database of airport weather observations

Airport weather observations (METAR's) are routinely made at all major airports on the every hour on the hour. Our database consists of a flat-file archive of 315,576 consecutive hourly weather observations from Halifax International Airport. [60] These observations are from the 36-year period from 1961 to 1996, inclusive. Based on the advice of a weather forecaster, we represent each hour with 12 selected attributes: 11 continuous attributes and 1 nominal attribute, precipitation, as shown in Figure 10.

Category	Attribute	Units
temporal	• date • hour	Julian date of year (wraps around) hours offset from sunrise/sunset
cloud ceiling and visibility	• cloud amount(s) • cloud ceiling height • visibility	tenths of cloud cover (for each layer) height in metres of ≥ 6/10^{ths} cloud cover horizontal visibility in metres
wind	• wind direction • wind speed	degrees from true north knots
precipitation	• precipitation type • precipitation intensity	"nil", "rain", "snow", etc. "nil", "light", "moderate", "heavy"
spread and temperature	• dew point temperature • dry bulb temperature	degrees Celsius degrees Celsius
pressure	• pressure trend	kiloPascal · hour $^{-1}$

Figure 10. Twelve attributes of an airport weather observation (METAR).

The file size is 6 Megabytes. The file is in a standard, column-delimited ASCII format, hence no preprocessing is needed. Very few values are missing and most of the reports appear to be reliable (i.e., plausible), hence no additional quality control is applied to the file prior to its use.

3.2 Fuzzy *k*-nn algorithm

This section explains how we built an expert-emulating, similarity-measuring fuzzy *k*-nn algorithm, or function. The similarity-measuring function, *sim*, is used to find *k* nearest neighbors. The function is given two cases, each identified by unique time indexes t_1 and t_2, and it returns a real number proportional to the degree of similarity of the two cases such that

---

[60] Halifax International Airport is located in Nova Scotia, Canada at coordinates 44º 53' North 63º 30' West at an elevation of 145 meters above sea level. The airport is situated 30 kilometers north from the Atlantic coast near the top of gently sloping terrain.

$$0.0 < sim(t_1, t_2) \leq 1.0$$

Because all weather cases are unique and because the value of *sim* is calculated to double precision, *sim* can identify *exactly k nearest neighbors*. There are no null search results and no ties.

The three steps to construct and use the algorithm are:

1. Configure similarity-measuring function.
2. Traverse case base to find *k*-nn.
3. Make prediction based on weighted median of *k*-nn.

The first step is performed only once and the second and third steps are performed every time weather prediction is made. All the algorithm design work is done in the first step. The steps are described in the following three subsections (3.2.1, 3.2.2, and 3.2.3).

3.2.1 Configure similarity-measuring function

To configure the function, an expert weather forecaster uses a fuzzy vocabulary to provide knowledge about how to perform case comparisons. A detailed sample questionnaire for knowledge acquisition is shown in Appendix A. The knowledge acquisition procedure is described in general terms in the following two subsections.

3.2.1.1 Expert specifies attributes to compare and the order in which they are to be compared

An expert suggests which attributes are important for matching. For the WIND-1 system, we selected the 12 attributes listed above in Figure 10.

An object is to summarily rule a case out of contention with the fewest possible number of tests. Hence, based on advice of the domain expert, tests most likely to discriminate are performed first. For instance, the expert suggests that the season attribute similarity be tested before the temperature attribute similarity. Winter cases are very unlike Summer cases, but a winter case and a summer case may both have temperatures of 10º Celsius. Therefore, date of the year is a better discriminator than temperature.

When cases are compared, their attributes are compared in the same order that they are listed in in Figure 10, namely: date of the year, hour of the day, cloud amount(s), cloud ceiling height, visibility, wind direction, wind speed, precipitation type, precipitation intensity, dew point temperature, dry bulb temperature, and pressure trend. When initial tests indicate great dissimilarity, subsequent tests are dispensed with.

3.2.1.2 Expert describes fuzzy relationships between attributes

This is the crucial step in system design. As the expert describes fuzzy relationships between attributes using fuzzy words, corresponding fuzzy sets are constructed to emulate expert case comparison. Appendix A shows a sample questionnaire for knowledge acquisition that a developer could ask a domain expert complete in order to obtain knowledge of how to evaluate degree of similarity between various comparable attributes of cases. Thereby, the expert imparts their sense of discrimination via fuzzy words into fuzzy sets, and we acquire knowledge about how to compare three kinds of attributes: ***c****ontinuous numbers*, ***a****bsolute numbers* (i.e., magnitude), and ***n****ominal attributes* (e.g., snow and rain) . And we acquire knowledge about how to weight recency, or in other words, how to ***f****orget older dissimilarities*. This acquired knowledge is represented below in four functions —μ^c, μ^a, μ^n, μ^f—in the following four sections in Figure 11, Figure 12, Figure 13, and Figure 14 (b).

Continuous-number attributes

For each continuous attribute, x_i, the expert specifies a value of c_i which is the threshold for considering two such attributes to be near each other. A fuzzy set is constructed accordingly as shown in Figure 11.

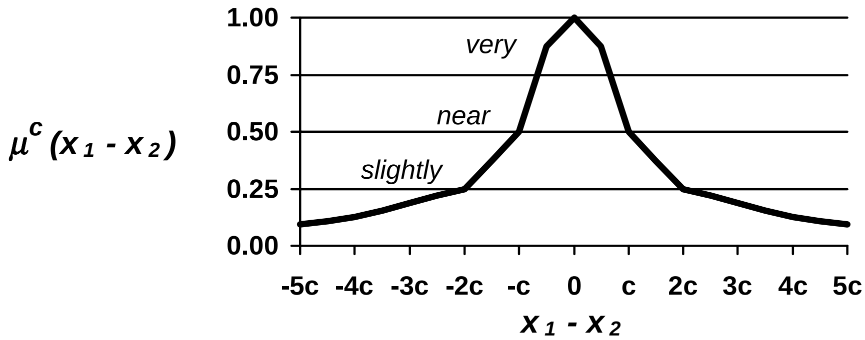


Figure 11. Fuzzy set for comparing continuous-number attributes. Similarity-measuring function emulates how the expert evaluates the degree to which continuous attributes are near each other. The expert specifies a value of c corresponding *near* such that $\mu(x_1 - x_2) \geq 0.50 \Leftrightarrow$ "x_1 is *near* x_2". Tails taper off asymptotically towards 0.0, such that $\mu(x) > 0.0$, which prevents null results from searches.

Comparing two homogeneous, continuous attributes with a fuzzy set, as shown in Figure 11, is a basic application of fuzzy sets. Multiple attributes of cases can be weighed collectively by aggregating the result of a set of such operations (e.g., taking the "max of the min"). Each fuzzy set enables the *sim* function to match based on individual attributes. As sets are added for multiple attributes, the *sim* function gains the ability to match more complicated cases.

Fuzzy sets such as the one shown in Figure 11 are used to compare the attributes of date of the year, hour of the day, wind direction, dew point temperature, dry bulb temperature, and pressure trend.

Date of the year and hour of the day are important temporal attributes because weather strongly correlates to seasonal and diurnal cycles. The closer these attributes of two cases are to each other, the more analogous the cases are. For example, two cases are considered *near* each other if they are within 30 Julian days of each other and their offsets from sunrise/sunset are within one hour of each other.

Absolute-number attributes

If attributes are limited to the zero-or-above range (e.g., absolute wind speed), then it is their relative magnitudes that are important for matching. Therefore, they are compared using a modified ratio operation, with special routines to handle for values near zero, as shown in Figure 12.

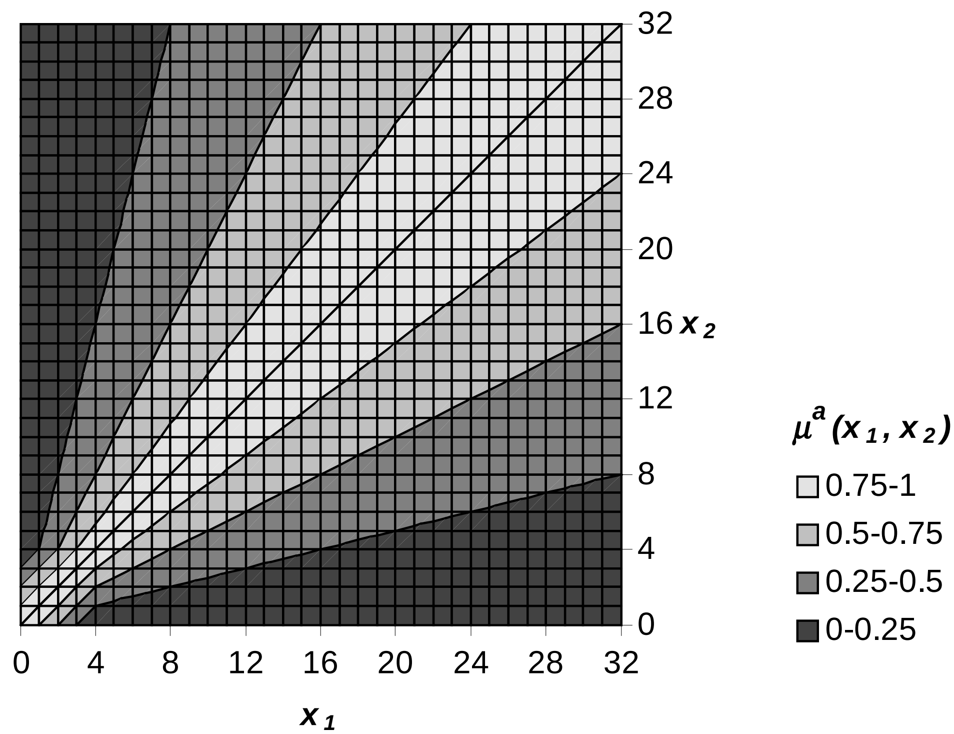


Figure 12. Fuzzy decision surface for comparing absolute-number attributes. Fuzzy similarity-measuring surface measures how similar two absolute values are to each other. The above surface determines the similarity of two wind speeds, x_1 and x_2, where speed is measured in knots. Wind speed values above 32 are truncated to 32.

The fuzzy decision surface shown in Figure 12 is used to compare the attribute of wind speed. Surfaces similar to the one shown in Figure 12 are used to compare the attributes of cloud amount(s), cloud ceiling height, and visibility.

Nominal attributes

To compare nominal attributes, such as precipitation type, a similarity measuring table (a symmetric matrix) is used of the form shown in Figure 13.

The table of fuzzy relationships shown in Figure 13 is used to compare the attribute of precipitation type. A table similar to that shown in Figure 13 is used to compare the attribute of precipitation intensity.

	Nil	Drizzle	Showers	Rain	Flurries	Snow	…
Nil	1.00	0.02	0.03	0.01	0.03	0.01	…
Drizzle	0.02	1.00	0.50	0.50	0.10	0.05	…
Showers	0.03	0.50	1.00	0.75	0.10	0.10	…
Rain	0.01	0.50	0.75	1.00	0.10	0.25	…
Flurries	0.03	0.10	0.10	0.10	1.00	0.75	…
Snow	0.01	0.05	0.10	0.25	0.75	1.00	…
…	…	…	…	…	…	…	…

$\boldsymbol{\mu^n(type_1, type_2)}$

Figure 13. Fuzzy relationships between nominal attributes. Similarity-measuring table measures how similar two nominal attributes are to each other.

Forget older attributes

The similarity of two cases is determined according to their newest and their most dissimilar attributes. [61] The older attributes are in compared cases—that is, the farther back in time they are from their respective time-zeroes— the less weight is accorded to their dissimilarity. In effect, this is the same as forgetting older attributes in comparing cases. Such a forgetting function is shown in Figure 14 (b). After two comparable attributes of two cases are

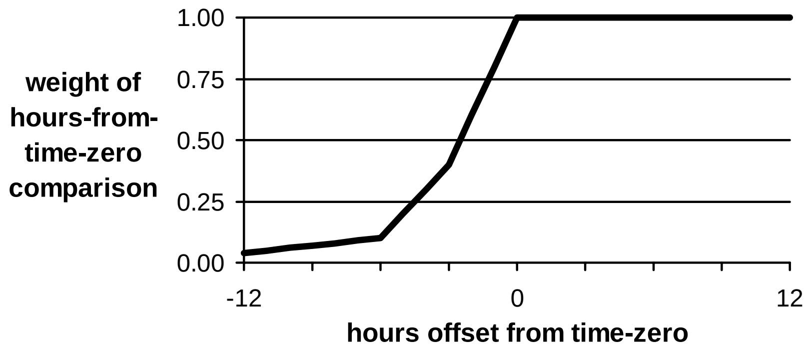
[61] Equating *highest similarity* with *lowest dissimilarity* harks back to the argument that a chain is only as strong as its weakest link, as explained in Section 2.2.3 on page 52.

compared to yield a similarity value, μ', the value of μ' is moderated using the forgetting function, such that

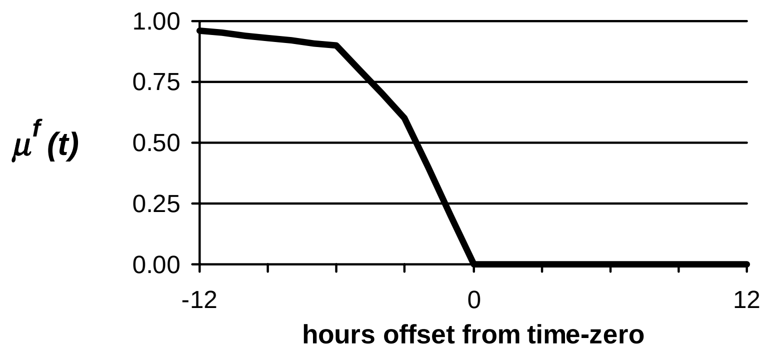
$$sim = max\ \{\mu', \mu^f(t)\}$$

So, with reference to Figure 14 (b), we see that 3-hour-old attributes can never imply *sim* < 0.6, whereas time-zero attributes or auxiliary predictors (with t > 0), can imply *sim* ≅ 0.0.

The more recent an attribute is, the more important it is for matching. Likewise, any auxiliary predictors, such as NWP, are important for matching. Each case has a temporal span of 24 hours composed of three parts: 12 *recent past* hours, 1 *time-zero* hour, and 12 *future* hours. (These three parts of a case are illustrated ahead in Figure 16, page 87.) The contributions to similarity measurement of cases, from corresponding hours of cases, are weighted to maximize the contribution of recent hours and to maximize the contribution of any available foreknowledge (e.g., prevision guidance from NWP), as shown in Figure 14.



(a) Recency weighting function. The newer attributes in compared cases are, the more their similarity is weighted. Greatest weight is given to recent attributes and future attributes, such as auxiliary predictors from NWP.



(b) Forgetting function. This function is the complement of the above recency weighting function, Figure 14 (a). The newer attributes in compared cases are, the stricter their similarity comparison. Corresponding hour-to-hour comparisons are "maxed" with the forgetting function. Thus, for example, comparison of attributes 3 hours prior to time-zero will result in similarity $\mu \geq 0.6$; whereas, comparison of attributes measured at time-zero, or predicted by other means for after time-zero, may result in similarity $\mu = 0.0$. Thus, the function emulates the forgetting of less relevant older attributes in case-to case comparison.

Figure 14. Fuzzy weighting for recency of attributes.

3.2.2 Traverse case base to find *k*-nn

After *sim* has been configured as explained above, it can be used to make weather predictions provided with a database of weather cases, a present incomplete weather case, and whatever foresight-offering guidance is available. Appendix B shows a worked-out, specific example of the procedure for making a prediction based on the fuzzy *k*-nn. The rest of this subsection illustrates the procedure generally.

First, for each hour of two cases being compared, the overall degree of similarity of their attributes is computed as the minimum value of μ for the compared attributes.

$$\mu_{hour\text{-}to\text{-}hour}(t) = min\{\mu^{c}_{1}, \ldots , \mu^{a}_{i}, \ldots , \mu^{n}_{m}\} \quad \text{for} \quad t = t_0\text{-}12 \ldots t_0$$

Second, the overall degree of the similarity of two cases is computed as the minimum value of all the hours' values of μ where each hour's value of μ is tempered by the "forgetting function."

$$\mu_{case\text{-}to\text{-}case}(t_0) = min\{max\{\mu_{hour\text{-}to\text{-}hour}(t_0\text{-}12),\mu^{f}(\text{-}12)\}, \ldots , max\{\mu_{hour\text{-}to\text{-}hour}(t_0),\mu^{f}(0)\}\}$$

The database of cases is represented conceptually in Figure 15. Each block represents an observation taken at one instant in time. Observations are taken at regular intervals. So, the series of blocks represents the time series.

$a_1(1)$	$a_1(2)$		$a_1(n)$
$a_2(1)$	$a_2(2)$	…	$a_2(n)$
:	:		:
$a_m(1)$	$a_m(2)$		$a_m(n)$

Figure 15. Structure of cases (i.e., weather observations). *m* attributes per observation, *n* observations. *Observation* is synonymous with *tuple*.

The parts of a temporal case are shown in Figure 16. A temporal case is a short segment of a long record of a multidimensional, real-world process—in our case, weather observations. Past cases in the database are complete temporal cases. The present case is an "incomplete temporal case"—it's missing the "future" part.

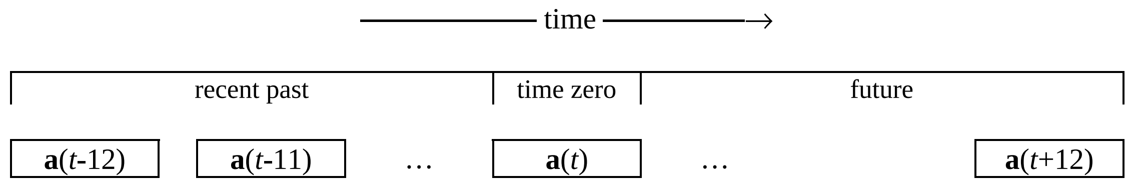


Figure 16. Temporal case. A series of weather observations centered on time *t*. Thus *t* is the specific case index. Series spans from 12 hours before to 12 hours after time *t*.

The time series of past cases is traversed and temporal cases are measured for degree of similarity with the present case using *sim* as shown in Figure 17.

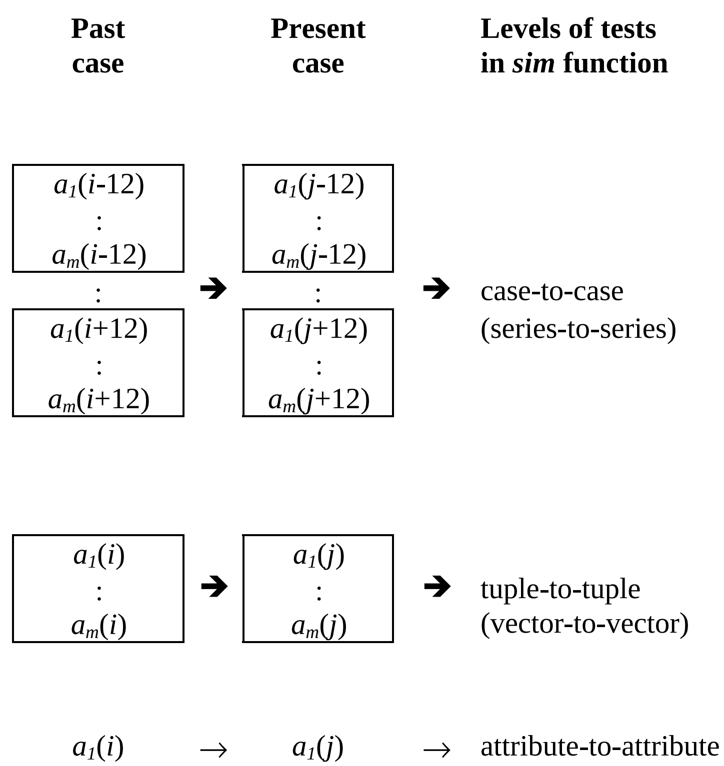


Figure 17. Temporal cases are compared in nested operations at three levels. Specific case indexes are *i* and *j*, the "time-zeros" of the cases being compared. The "⟶" symbol denotes a comparison of two attributes, a fuzzy operation of one of the four kinds described above in subsection 3.2.1.2. The "➔" symbol denotes an aggregation of numerous such fuzzy comparisons (e.g., max of the min).

The overall similarity of two temporal cases is equal to the lowest similarity of any comparable elements (i.e., equal to the lowest attribute-to-attribute similarity). The *sim* function performs tests progressively, following the efficiently discriminating sequence advised by the expert (subsection 3.2.1.1). The similarity measuring operation is halted if the similarity falls below the α-level (i.e., the current minimum threshold to join the analog set). As better and better analogs are collected during the traversal, the α-level rises accordingly. As the α-level rises, the case-to-case comparison can be aborted after fewer and fewer attribute-to-attribute comparisons. Thus, although the above 3-level-deep case comparison algorithm is potentially of Order(n^3) complexity, in practice we expect it to be much closer to Order(n) complexity.

The analogs are collected and α is recorded in a linked list data structure as shown in Figure 18.

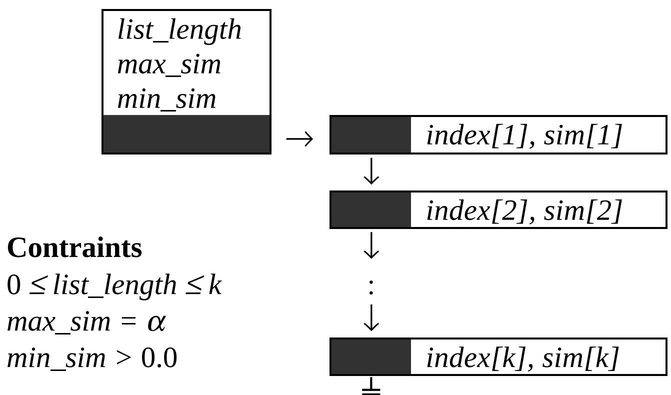


Figure 18. Linked list of *k*-nn, or weather analogs. The similarity of each case is represented by *sim[i]*. By design, $0.0 < sim[i] \leq 1.0$. The least similar member in the set has overall similarity equal to *sim[k]*, thus α-*level* = *sim[k]*.
list_length is the number of cases in the list.
max_sim and *min_sim* describe the similarity values of the most similar and least similar cases in the list respectively.
index[i] identifies the time of the temporal case.

3.2.3 Make prediction based on weighted median of *k*-nn

The predictands are cloud ceiling and visibility. Predictions are made by weighted median of the *k* most analogous cases, the *k*-nn. Each case is weighted according to its similarity to the present case: $0.0 < sim[i] \leq 1.0$.

A worked-out example of how the fuzzy *k*-nn algorithm measures the similarity of two temporal cases with a present case and makes predictions based on the fuzzy *k*-nn is shown in Appendix B.

3.3 Comparison to previous approaches

This section compares our fuzzy *k*-nn algorithm with a classic fuzzy nearest prototype algorithm and with a classic model of CBR.

3.3.1 Fuzzy nearest prototype algorithm

Compare our fuzzy nearest prototype algorithm in Figure 19 and Figure 20 with that of Keller et al. (1985), shown in Figure 8 (page 60).

Let $W = \{A_1, A_2, \ldots, A_n\}$ be the set of n past cases representing n potential analogs.

```
//  x is the present case
main(x)
{
    α = 0.0
    list_lenth = 0

    //  traverse past case base of n cases searching for analogs
    for i = 1 to n
    {
        //  test all past cases for admissibility into the k-nn set
        //  α is the admission threshold
        //  if similarity of new case > α
        //  then new case is saved and α rises accordingly
        if (sim(A_i, x) > α)
        {
            α = sim(A_i, x)

            //  Update the linked list of k-nn, shown in Figure 18 on page 88
            insert_to_ordered_k-nn_list(A_i)

            //  limit list length to k
            list_length = list_length + 1
            if list_length > k
            {
                remove_from_ordered_k-nn_list_least_similar_member()
                list_length = list_length - 1
            }
        }
    }
    return(weighted median of k-nn)
}
```

where $sim(A_i, x)$ is represented by the function shown in Figure 20.

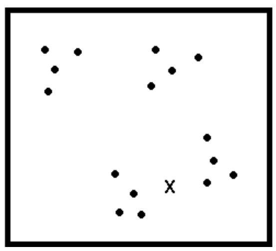
Figure 19. Cyclic algorithm for *WIND-1* in pseudocode. Nearest prototype algorithm centered on the present case and with α-level progressively rising as case base is traversed and better analogs accumulate in ordered k-nn list. This algorithm contrasts with that of Keller et al. (1985), which is shown in Figure 8 on page 60.

```
double sim(A_i, x)
{
    // result = 1.0 implies similar, result = 0.0 implies dissimilar
    result = 1.0

    // traverse series from most recent tuple to least recent tuple
    // h is the number of hours in a temporal case
    for h = tuples_in_series down to 1)
    {
        for (m = 1 to attributes_in_tuple)
        {
            // μ_m is an attribute-specific fuzzy comparison operation,
            // i.e, μ^a, μ^c, and μ^n as shown Figure 11, Figure 12, and Figure 13
            // on pages 81–83
            result = μ_m(A_i[h, m], x[h, m])

            // Reduce weight of old attributes in case
            // result approaches 1.0 for non-recent attributes
            // In effect, calculate maximum of result with
            // of fuzzy set shown in Figure 14 (b) on page 85.
            weight_for_recency(result)
            if (result < α)
            return (0.0)
        }
    }
    return(result)
}
```

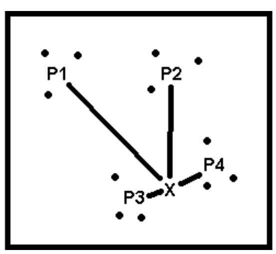
Figure 20. Similarity-measuring function: *sim*.

Our algorithm differs from that of Keller et al. (1985) mainly in the following way: Rather than calculating the distance from the new case to every selected prototypical case and then, accordingly, calculating the degrees of membership of the new case in all the prototypical categories as (Keller et al. 1985) does, our algorithm calculates the degrees of similarity between the present case and its k nearest neighbors. Only the k instances of A_i that are nearest to the present case are assigned values for degree of similarity to the present case. Our approach is, in a sense, a reversal of that of Keller et al.: Rather than calculating the degrees of membership of the present case in selected categories, which are based on prototypes, we calculate the degrees of membership of past cases in the one-off "category" that is based on the present case.

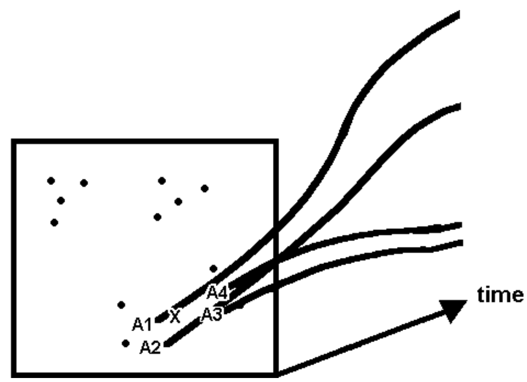
As the case base is traversed, and each prospective analog A_i is subjected to a series of similarity tests, cases are either summarily ruled out of contention or finally ruled into the k-nn set. As the case base is traversed, and better and better analogs are collected, the α- level rises accordingly. This strategy enables computational savings. Because we are only interested in identifying exactly *k* nearest neighbors, most of the past cases can be summarily ruled out of contention. Rather than fully calculating the similarities between a new case and all predefined prototypes, as illustrated in Figure 21 (b), it predicts the outcome of the new case based on a few nearest neighbors, as illustrated in Figure 21 (c).



(a) Points represent past cases.
X represents new case.



(b) New case *X* is classified according to distance from selected prototypes, *P1-P4*.



(c) New case is predicted for based on outcome of nearest neighbors, *A1-A4* (i.e., the "analog ensemble.")

Figure 21. Classification based on prototypes contrasted with prediction based on nearest neighbors. (a) A number of fully-described points and one new partly-described point labelled as *X*. (b) Classification of a new case based on selected prototypes. (c) Prediction for a new case based on *k* nearest neighbors.

3.3.2 Classic case-based reasoning

Compare our case-based reasoning flowchart, shown in Figure 22, with that of Riesbeck and Schank (1989), shown in Figure 1 (page 4). The differences between classic CBR and fuzzy CBR are listed in Figure 23.

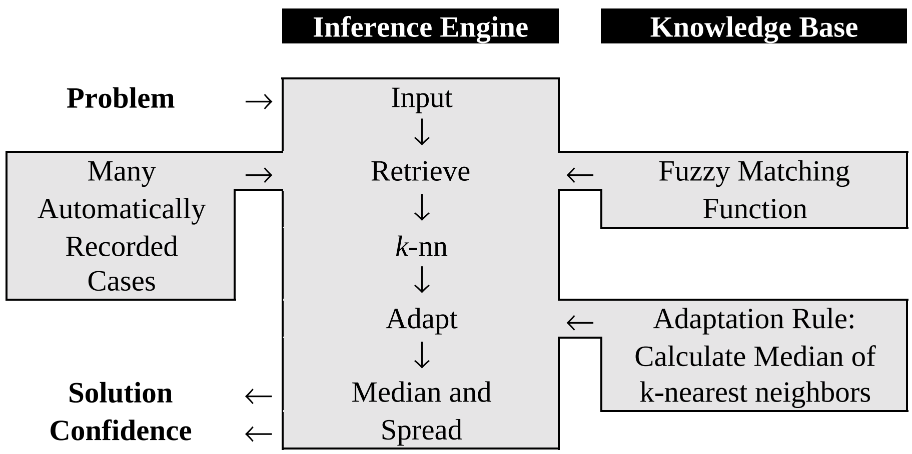


Figure 22. Fuzzy case-based reasoning. This flowchart contrasts with that of Riesbeck and Schank (1989), which resembles the one shown in Figure 1 (page 4). The case authoring problem is avoided by using many automatically-recorded cases and a fuzzy matching function. The case adaptation problem is avoided by using many automatically-recorded cases (with supposed good analogs) and by simply calculating a median of k-nearest neighbors.

Classic CBR	Fuzzy CBR
• Uses abstract indexing rules to determine classes.	• Describes cases with their numerical dimensions and nominal types.
• Revises the case memory with tested solutions.	• Uses a large case base of continuously accumulating actual cases.
• Attempts to repair solutions in a potentially endless loop.	• Bases solution on a weighted median of numerous similar cases and associates a confidence in the solution based on the spread of those cases.
• Perform the series of operations: Proposed Solution → Test → Failure Description → Explain → Predictive Features → Indexing Rules.	• Obtains knowledge about predictive features through knowledge acquisition from domain expert who explains similarity with fuzzy words.

Figure 23. Differences between classic CBR and fuzzy CBR.

4. Experiments

The aim of our experiments is to test our hypothesis. [62] The previous chapter described the WIND-1 system, its components, and how it is configured. The algorithm for WIND-1 (shown in Figure 19 on page 90) is coded in C and implemented on a Hewlett Packard 9000 series workstation.

In this chapter, we evaluate the performance of the WIND-1 system by conducting a series of experiments, as is common current practice in machine learning algorithm validation, rather than simply by determining the plausibility of the results, as used to be common practice (Langly 1996).

This chapter describes a series of experiments that test the prediction accuracy of the WIND-1 system. Results of WIND-1 are measured using standard tests of prediction accuracy, presented in the form of graphs, and interpreted.

Results are presented in five sections. In section 4.1, we vary the attribute set in order to determine the relative predictive value of various predictors. In section 4.2, we vary the number of analogs used to make forecasts (*k*) in order to evaluate the tradeoff between using a smaller number more analogous cases and a larger number of less analogous cases and, thereby, select a good value for *k*. In section 4.3, we vary the size of the case base in order to assess the importance of having a large case base. In section 4.4, we decrease the level of fuzziness in the fuzzy sets in the fuzzy *k*-nn algorithm in order to assess the importance of fuzziness itself in similarity measurement And finally, in section 4.5, we test the prediction accuracy of WIND-1 against a benchmark prediction method, persistence.

Experiment design

Each experiment consists of a forecasting scenario. Five sets of experiments are conducted. In each set of experiments we systematically change the fixed parameters of WIND-1 and measure the resultant effects on forecast accuracy. The fixed parameters (independent variables) are: the attribute set, the number of analogs used to make forecasts, the size of the case base, and the fuzzy membership functions (i.e., level of fuzziness in the similarity-measuring sets). The output (dependent variables) are, for each individual forecast, forecast values of cloud

[62] Hypothesis: Querying a large database of weather observations for past weather cases similar to a present case using a fuzzy k-nearest neighbors algorithm that is designed and tuned with the help of a weather forecasting expert can increase the accuracy of predictions of cloud ceiling and visibility at an airport.

ceiling and visibility, and, for each set of experiments, a summary of the accuracy of all the forecasts made.

The first four sets of experiments serve a dual purpose. First, they test the contribution of individual components of the system. Second, they suggest how to adjust these components in order to maximize the accuracy of the system. The last set of experiments pits the system against a competitive prediction technique, persistence forecasting. [63]

In most of the sets of experiments, the first 35 years of weather data (1961-1995) is used as the case base and the final year of data (1996) is used as a source of "new cases." Such data segregation prevents sharing of information between past and new cases. In one set of experiments (Section 4.3), however, we test the effect of reducing the size of the case base.

In each set of experiments, 1000 hours are chosen at random from the 1996 weather archive and are each used as an hour to produce a forecast for. So, in each set of experiments, 1000 simulated forecasts are produced. For purposes of comparison, the same 1000 randomly-chosen hours are used in each set of experiments. This is a control so that the effect of varying other input can be tested.

In each individual experiment, a case is taken from the 1996 data and is used as a present case. It is input to WIND-1. During the forecast process, the outcome of the present case is hidden from WIND-1. WIND-1 produces a forecast for the present case based on the outcomes of the *k*-nn in the case base, the *k* most analogous past cases for the present case. After the forecast process, the accuracy of the forecast is verified by comparing the forecast with the then unhidden outcome of the present case. [64]

---

[63] To be useful for airport weather prediction, a system should produce results more accurate than the results of persistence forecasting. To forecast persistence, one simply takes the known values of ceiling and visibility at time-zero, the beginning time of a forecast period, and assumes that they will not change through the forecast period,

Dallavalle and Dagastaro (1995) compared the skill of persistence-based forecasts with the skill of forecasts produced locally by the National Weather Service and found that "persistence forecasts appeared to have higher skill than the local forecasts for the 3-hour projection." When skill was considered for the six-hour projection, neither method was clearly superior; persistence had a higher Critical Success Index, but locally produced forecasts had a higher Heidke skill score.

[64] During case-to-case comparison, we hid the outcomes of the present case so as not to "contaminate" our results. We did not give WIND-1 any prevision of attributes to guide its search for analogs. However, we envision that a future operational version of this system (WIND-2) will incorporate available prevision of attributes (e.g., imminent wind shifts or precipitation onsets predicted by other means) to guide its search for analogs.

Verification method

Each forecast is verified using standard measures of weather forecast accuracy, measures that are described in detail by Stanski et al. (1999), and summarized as follows.

Forecasts are verified according to the accuracy of forecasts of three significant flying categories, categories that are defined in Figure 24.

ceiling (m)		visibility (km)		flying category
< 200	or	< 3.2	⇒	below alternate
≥ 200	and	≥ 3.2	⇒	alternate
≥ 330	and	≥ 4.8	⇒	VFR

Figure 24. Flying categories.

Three sorts of prediction-versus-actual outcomes are counted: *hits*, *false alarms*, and *misses*. If an event is predicted and it occurs, it counts as a hit. If an event is predicted and it does not occur, it counts as a false alarm. If an event is not predicted and it does occur, it counts as a miss. How outcomes of forecast and observed events are classified is shown in Figure 25.

		OBSERVED	
		YES	NO
FORECAST	YES	hit	false alarm
	NO	miss	non-event

Figure 25. How outcomes of forecast and observed events are classified.

From the frequencies of these outcomes, three meteorological statistics are calculated: Reliability (i.e., Frequency of Hits, or FOH), Probability of Detection (POD), and False Alarm Ratio (FAR). Values are calculated as shown in Figure 26.

Reliability =	FOH =	hits / (hits + false alarms)
Probability of Detection =	POD =	hits / (hits + misses)
False Alarm Ratio =	FAR =	false alarms / (hits + false alarms)

Figure 26. Formulae for verification of forecasts. Values of hits, false alarms, and misses are the summed outcomes of 1000 simulated forecasts. High levels of prediction accuracy are indicated by high Reliability, high Probability of Detection, and low False Alarm Ratio.

To ensure that WIND-1 verified its forecasts correctly, we double-checked the verification results. We had WIND-1 conduct a small set of experiments, based on only 10 forecasts, verified the forecast accuracy manually, and compared our manually-generated results with WIND-1's automatically-generated results. The results were the same.

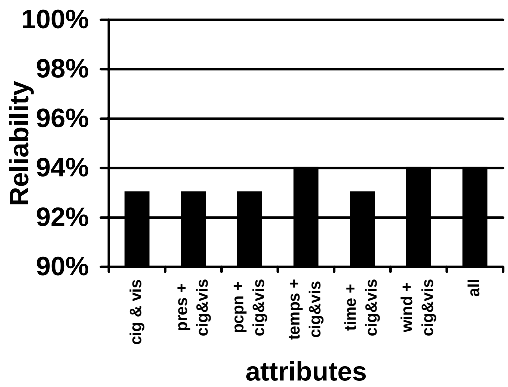
The purpose of the first set of experiments is to determine the relative predictive value of various attributes (i.e., predictors) which are recommended by a forecasting expert. Seven sets of such attributes are listed in Figure 27.

Set number	**Abbreviation**	**Attribute set**
1	*cig* & *vis*	cloud ceiling and visibility (cig & vis)
2	*pres* + *cig&vis*	pressure tendency, cig & vis
3	*pcpn* + *cig&vis*	precipitation type and intensity, cig & vis
4	*temps* + *cig&vis*	dry bulb temperature and dew point temperature, cig & vis
5	*time* + *cig&vis*	offset from sunrise/sunset and date of year, cig & vis
6	*wind* + *cig&vis*	wind direction and speed, cig & vis
7	*all*	all of the above

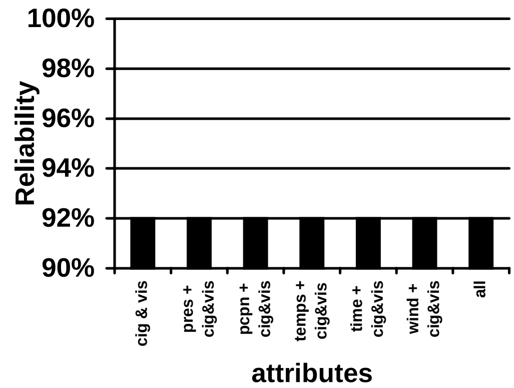
Figure 27. Attribute sets for matching. All attribute sets include cloud ceiling and visibility (cig & vis). The first set consists of only ceiling and visibility, the next five sets each consist of ceiling and visibility plus one other type of attribute, and the last set consists of all available attributes.

Because our objective is to predict ceiling and visibility, and because ceiling and visibility are known to be strongly autocorrelated, the attributes of ceiling and visibility are included in each attribute set. Figure 28 presents the results of varying the attribute set.

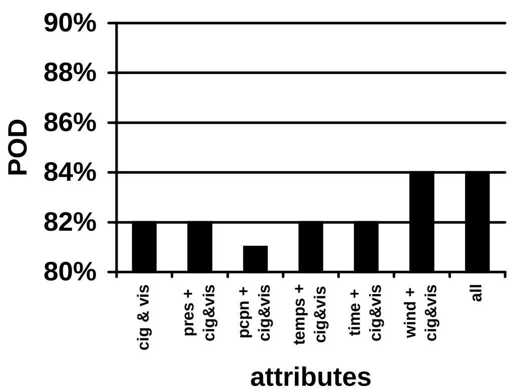
4.1 Effect of varying attribute set



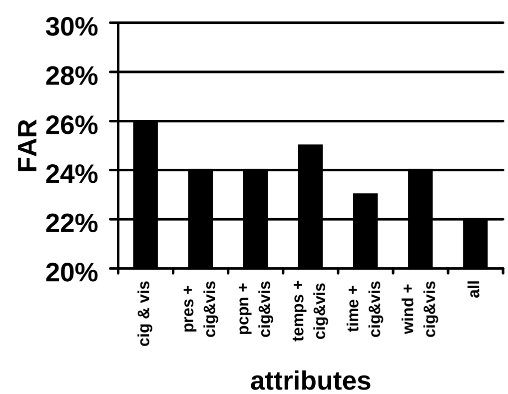
(a) Reliability of "alternate"



(b) Reliability of "VFR"



(c) Probability of Detection of "below alternate"



(d) False Alarm Ratio of "below alternate"

Figure 28. Effect of varying attribute set. Graphed values are average accuracy of 0-to-6-hour predictions. System configuration: k =16, length of case base = 35 years.

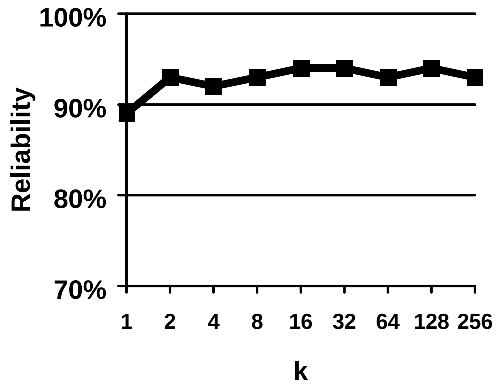
The benchmarks for accuracy in each of the charts in Figure 28 are the left-most bars in each graph, the accuracy resulting from matching cases based only on their cloud ceiling and visibility attributes.

As attributes are added to the similarity measurement, the resulting accuracy of the forecasts tends to increase.

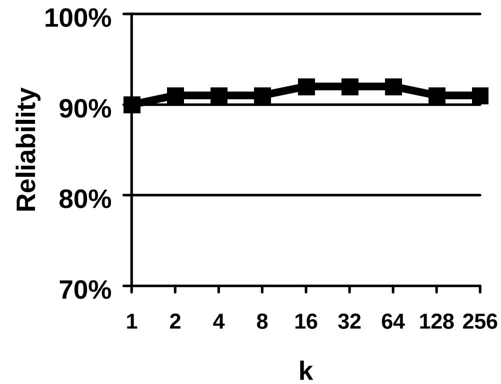
The best combination of attributes tested is the complete set of available attributes, the right-most bars in each graph. This combination results in the lowest False Alarm Ratio of below alternate (Figure 28 (d)) and ties for the highest values of probability of detection of below alternate, reliability of alternate, and reliability of VFR (Figure 28 (a), (b), and (c)).

The reliability of VFR is barely affected by the combination of attributes used (Figure 28 (b)). This is probably due to the climatological preponderance of VFR conditions in weather (conditions that are specified by cloud ceiling and visibility) and the strictness of the matching in the cloud ceiling and visibility conditions. VFR is the largest cluster of weather conditions and it is defined by the attributes of ceiling and visibility, so the contribution of other attributes is relatively small. In other words, it takes little skill to forecast the persistence of a most common condition: primary details (cloud ceiling and visibility) are sufficient to make accurate forecasts, and secondary details add little.

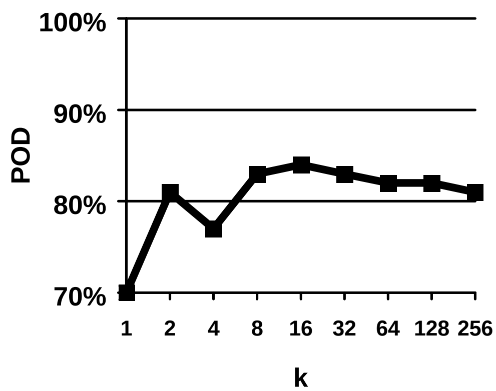
4.2 Effect of varying *k*



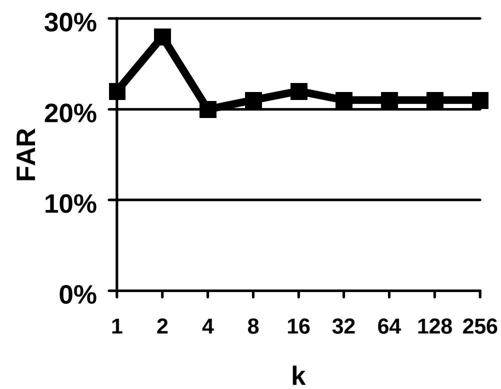
(a) Reliability of "alternate"



(b) Reliability of "VFR"



(c) Probability of Detection of "below alternate"



(d) False Alarm Ratio of "below alternate"

Figure 29. Effect of varying *k*. Predictions are based on weighted median value of *k* nearest neighbors. Graphed values are average accuracy of 0-to-6-hour predictions. System configuration: length of case base = 35 years.

The purpose of this experiment was to assess the effect of varying the number of analogous past cases used to make forecasts ("*k*" in the expression "fuzzy *k*-nn") in order to evaluate the tradeoff between using a smaller number more analogous cases and a larger number of less analogous cases, and thereby select a good value for *k*. We systematically varied the value of *k* and the results of this experiment are shown in Figure 29.

There is a peak in accuracy resulting from using 16 nearest-neighbor cases to form predictions, or about 0.005% (16 / 300,000) of all the available cases (Figure 29). This suggests the fuzzy *k*-nn algorithm is effective at identifying and ranking analogous cases; on average, the 16 nearest neighbors are more analogous and (and thus better bases for prediction) than the 256 nearest neighbors.

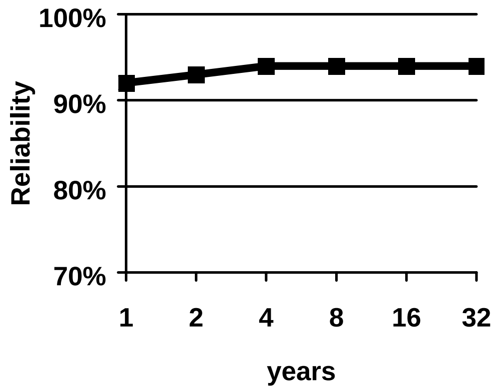
Accuracy tends to decrease as *k* decreases from 16 to 1. This suggests that it is more effective to base forecasts on small set of analogs than it is to base forecasts on single best analog. A similar effect is observed with “ensemble forecasting” technique in the field of numerical weather prediction in meteorology. [65]

All the graphs show a peak for *k*=2. You might wonder why a peak would occur in all four charts (recall that that high values of reliability and probability of detection imply high accuracy and high values of false alarm ratio imply low inaccuracy). The reason is that reliability, probability of detection, false alarm ratio tend to rise together because of the way they are formulated. In practice, forecasters try achieve a balance between high reliability, high probability of detection, and low false alarm ratio. [66]

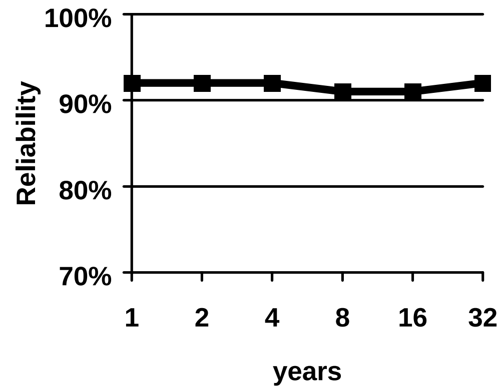
[65] Sivillo et al. (1997) define an *ensemble forecast* as: “a collection (an ensemble) of [numerical-weather-prediction-based] forecasts that all verify at the same time. These forecasts are regarded as possible scenarios given the uncertainty associated with forecasting.” Ensembles are made by running competitive and/or slightly-differently-initialized numerical weather prediction models concurrently. Sivillo et al. (1997) explain how the average of an ensemble of forecasts tends to be more accurate than single-scenario forecasts.

[66] To maximize reliability of forecasting and the probability of detection of an event, one simply needs to forecast the event to happen every time. The drawback with this strategy is that the false alarm ratio would rise at the same time. Forecasters refer to this strategy as "crying wolf." If one always raises an alarm, one will never miss an event, but one will also raise too many false alarms.

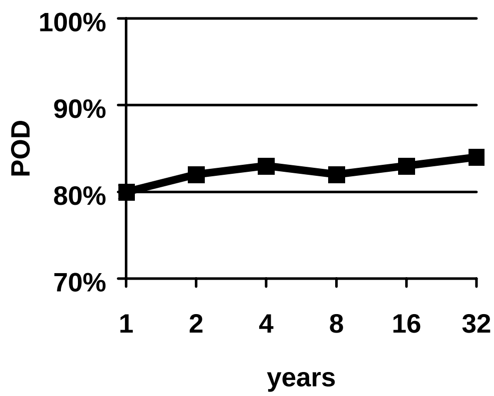
4.3 Effect of varying size of case base



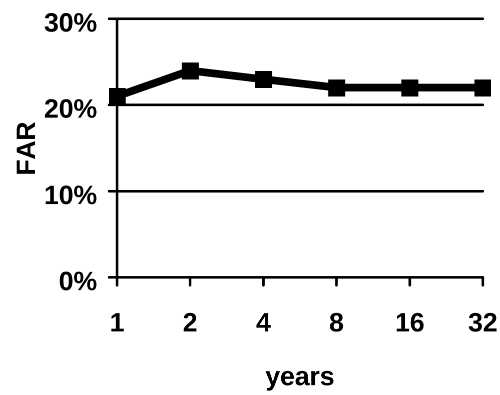
(a) Reliability of "alternate"



(b) Reliability of "VFR"



(c) Probability of Detection of "below alternate"



(d) False Alarm Ratio of "below alternate"

Figure 30. Effect of varying size of case base. Graphed values are average accuracy of 0-to-6-hour predictions. System configuration: k = 16.

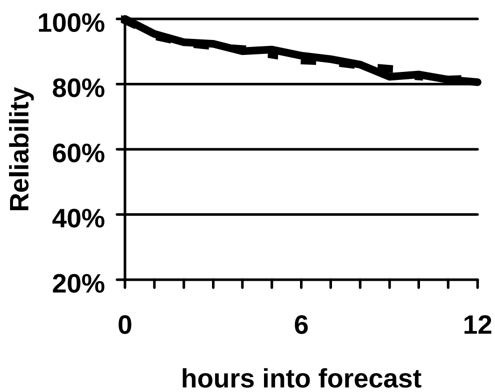
The purpose of this experiment is to determine effect of varying the size of the case base in order to assess the importance of having a large case base. As the size of the case base increases, supposedly, more and more potential good analogs are available for the fuzzy *k*-nn algorithm upon which to base predictions. This experiment addresses the question: "Is the fuzzy *k*-nn predictions method effective with a small case base, or does it require a large case base?"

This question is of practical importance because sizes of weather archives vary greatly from one airport to another. The size of the case base is varied and the results are shown in Figure 30.

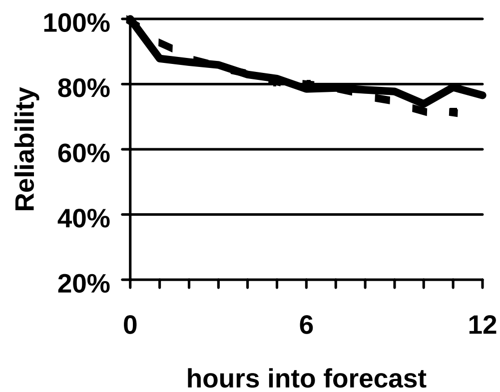
Accuracy generally rises as the size of the case base size increases from 1 year to 32 years, although there appears to be a slight dip in accuracy for a case base size of 8 years (Figure 30 (b) and (c)). The general rise in accuracy suggests that having a large case base is beneficial. The slight dip in accuracy for a case base size of 8 years—though probably insignificant at 1%—may suggest that, for the purposes of predicting for weather situations in the year 1996, the four-year period 1992-1995 contains a higher proportion of good analogs than the 8-year period 1988-1995.

Significantly, the relatively high accuracy with a case base size of 4 years suggests that the WIND-1 system could be useful for predicting at airports with relatively small weather archives. Most airports have recorded weather for at least 4 years.

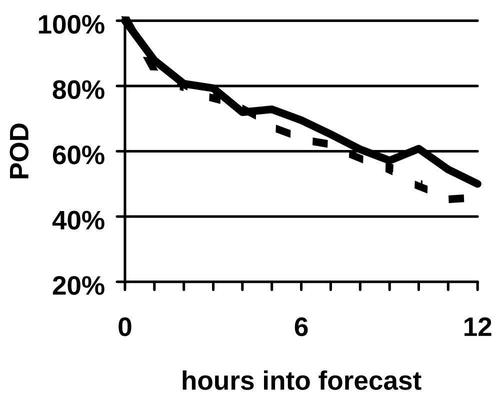
4.4 Effect of varying fuzzy set membership function



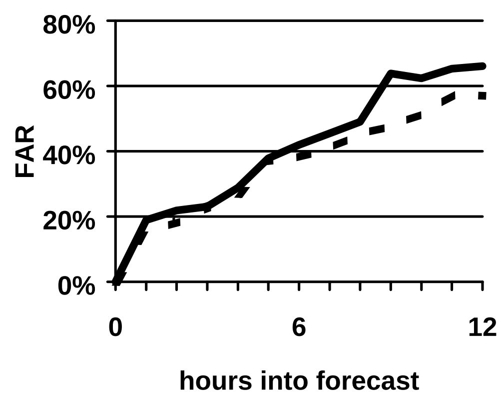
(a) Reliability of "alternate"



(b) Reliability of "VFR"



(c) Probability of Detection of "below alternate"



(d) False Alarm Ratio of "below alternate"

Non-fuzzy *k*-nn

Simple persistence

Figure 31. Effect of varying fuzzy set membership function. Fuzziness is eliminated by converting elicited fuzzy sets into crisp sets. Graphed values are accuracy of prediction for each hour in the 0-to-12-hour projection period. System configuration: k=16, length of case base = 35 years.

The purpose of this set of experiments is to test the effect of using fuzzy sets in the similarity-measuring function, as opposed to using non-fuzzy sets. When we propose to use fuzzy sets in a similarity-measuring algorithm, we are often asked: "Why not use *non-fuzzy* category based similarity measures?" This set of experiments addresses that question by substituting non-fuzzy (i.e., crisp) sets for the previously used fuzzy sets. To eliminate fuzziness, we modify the function μ_{fuzzy}, as follows.

$$\begin{aligned}&\textit{if } \mu_{fuzzy} < 0.5 \textit{ then } \mu_{crisp} = 0.0\\ &\textit{else } \mu_{crisp} = 1.0.\end{aligned}$$

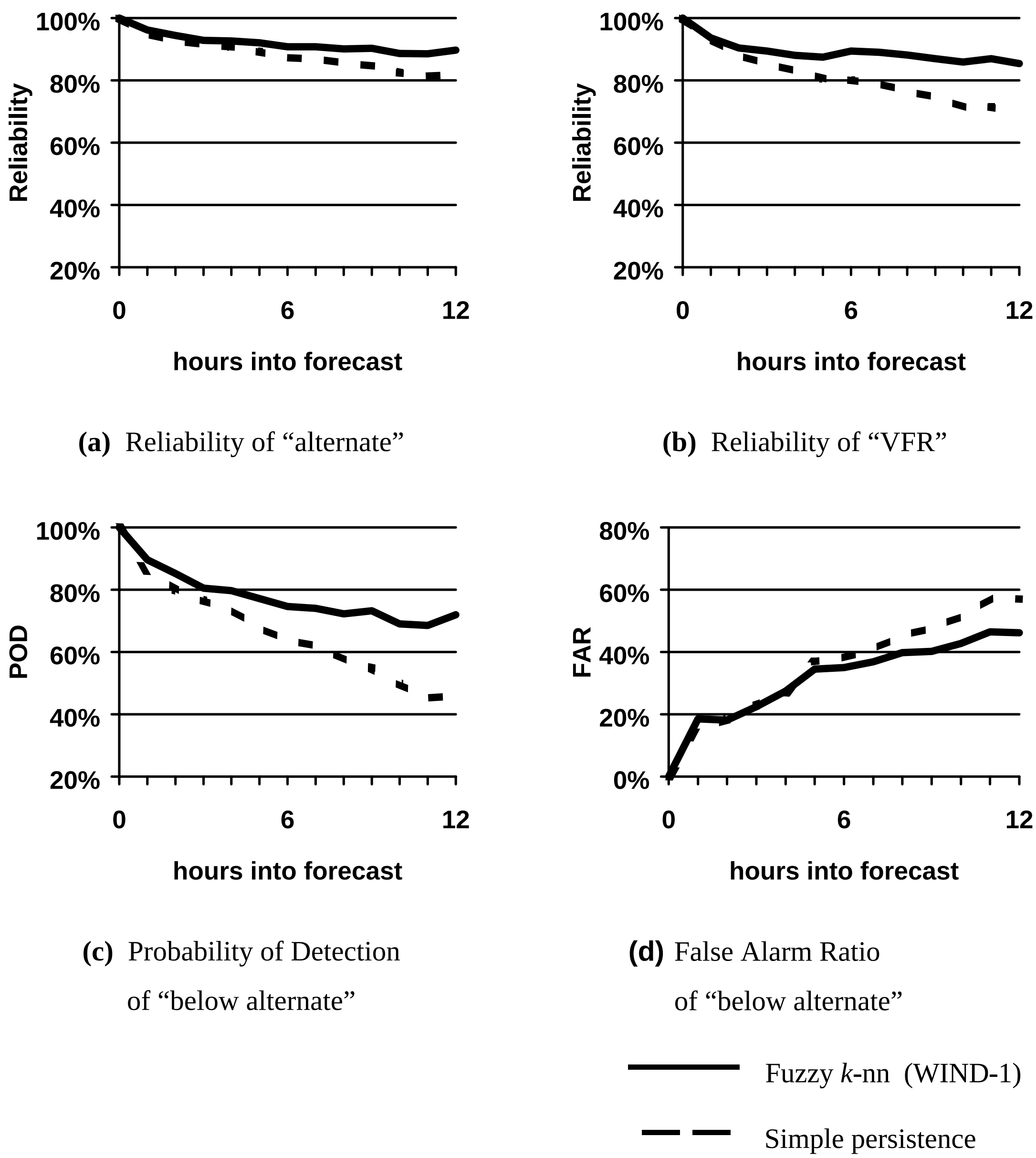
Thus, for two cases, every attribute at every hour compared yields a similarity measure equal to 0 or 1. Then the overall similarity of the cases is determined by taking the average of all the attributes values. The results are shown in Figure 31.

Non-fuzzy *k*-nn based predictions are slightly more accurate than simple persistence based forecasts in terms of reliability and probability of detection (Figure 31 (a), (b) and (c)), and slightly less accurate in terms of false alarm ratio (Figure 31 (d)). The results supports what aviation weather forecasters commonly believe: It is difficult to beat simple persistence (forecasting "no change") in the short-term.

The non-fuzzy *k*-nn prediction scheme is essentially a form of conditional-persistence-based forecasting. [67] For cases to be in the nearest-neighbor set of the case being forecast for, the condition is that their attributes must fall within discrete, plus-or-minus ranges of the attributes of the case being forecast for. The results suggest that it is even difficult for conditional persistence to beat simple persistence.

[67] Conditional-persistence based forecasting is referred to in meteorology as "climatological persistence" (Vislocky and Fritsch 1997).

4.5 System versus persistence



(a) Reliability of “alternate”

(b) Reliability of “VFR”

(c) Probability of Detection of “below alternate”

(d) False Alarm Ratio of “below alternate”

Figure 32. Accuracy of system compared to benchmark technique, persistence. Graphed values are accuracy of prediction for each hour in the 0-to-12-hour projection period. System configuration: k=16, length of case base = 35 years.

The purpose of this experiment is to compare the prediction accuracy of WIND-1 with that of the benchmark prediction method, persistence. The results are shown in Figure 32.

Fuzzy *k*-nn based predictions are significantly more accurate than simple persistence based forecasts in terms of reliability and probability of detection (Figure 32 (a), (b) and (c)), and generally more accurate in terms of false alarm ratio (Figure 32 (d)).

Fuzzy *k*-nn based predictions are significantly more accurate than non-fuzzy based predictions (compare Figure 32 with Figure 31). The only variation between the two experimental setups is the nature of the membership functions used to compare attributes. The fuzzy *k*-nn method uses fuzzy membership functions that span certain ranges around the case being forecast for; whereas, the non-fuzzy method uses 0-1-0 functions centered across the same ranges. This suggests that, compared to the accuracy of simple persistence, the significantly higher accuracy of fuzzy *k*-nn based forecasts is attributable to the use of fuzzy sets to measure similarity as opposed to using crisp sets. To the best of our knowledge, all previous methods used to measure similarity between weather cases have used only crisp sets.

5. Conclusion

Based on our literature review, experiments, and the results presented in the previous chapter, we conclude that querying a large database of weather observations for past weather cases similar to a present case using a fuzzy *k*-nearest neighbors (fuzzy *k*-nn) algorithm that is designed and tuned with the help of a weather forecasting expert can increase the accuracy of predictions of cloud ceiling and visibility at an airport.

We have proposed, implemented, and tested a fuzzy *k*-nn based prediction system called WIND-1. Its unique component is an expertly-tuned fuzzy *k*-nn algorithm with a temporal dimension. We tested it with the problem of producing 6-hour predictions of cloud ceiling and visibility at an airport given a database of over 300,000 consecutive hourly airport weather observations (36 years of record). Its prediction accuracy was measured with standard meteorological statistics and compared to a benchmark prediction technique, persistence. In realistic simulations, WIND-1 was significantly more accurate. WIND-1 produced forecasts at the rate of about one per minute.

The fuzzy *k*-nn based prediction method is significantly more accurate than the non-fuzzy based prediction method. The only variation between the two methods is the nature of the membership functions used to compare attributes of cases. The fuzzy *k*-nn method uses fuzzy membership functions that span certain ranges around the case being forecast for, whereas the non-fuzzy method uses 0-1-0 functions centered across the same ranges. This suggests that, compared to the accuracy of simple persistence, the significantly higher accuracy of fuzzy *k*-nn based forecasts is attributable to the use of fuzzy sets to measure similarity as opposed to using crisp sets. To the best of our knowledge, all previous methods used to measure similarity between weather cases have used only crisp sets.

Of significance to case based reasoning: We have shown how fuzzy logic can impart to case-based reasoning the perceptiveness and case-discriminating ability of a domain expert. The fuzzy *k*-nn technique described in this thesis retrieves similar cases by emulating a domain expert who understands and interprets similar cases. The main contribution of fuzzy logic to case-based reasoning is that it enables us to use common words to directly acquire domain knowledge about feature salience. This knowledge enables us to retrieve a few most similar cases from a large temporal database, which in turn helps us to avoid the problems of case adaptation and case authoring.

The fuzzy *k*-nn algorithm, even though it is of approximate Order(*n*) complexity, makes superior predictions with practical speed—with less than one minute of computation. This speed is achieved by strategically ordering the steps in a case-to-case similarity-measuring test and by stopping any test as soon as a step reveals that a case is dissimilar enough to be ruled out of the *k*-nn set without the need for further tests. For example, suppose we have a database of *n* past temporal cases. And suppose each case is described by *m* attributes and is *p* time units long, thus each case is described by $m \cdot p$ attributes. To measure the similarity of *every* past case, we would need to perform $n \cdot m \cdot p$ individual tests. However, we are only interested in finding the *k* most similar cases, and most cases can be ruled out of contention with a single test. So, the number of tests we need to perform is much closer to the order of *n* than it is to the order of $n \cdot m \cdot p$.

Of significance to meteorology and the aviation industry: Such a fuzzy *k*-nn weather prediction system can improve the technique of persistence climatology (PC) by achieving direct, efficient, expert-like comparison of past and present weather cases. PC is a sort of analog forecasting technique that is widely recognized as a formidable benchmark for short-range weather prediction. Previous PC systems have had two built-in constraints: they represented cases in terms of the memberships of their attributes in predefined categories and they referred to a preselected combination of attributes (i.e., defined and selected before receiving the precise and numerous details of present cases). The proposed fuzzy *k*-nn system compares past and present cases directly and precisely in terms of their numerous salient attributes. The fuzzy *k*-nn method is not tied to specific categories nor is it constrained to using only a specific limited set of predictors. Such a system for making airport weather predictions will let us tap many, large, unused archives of airport weather observations, ready repositories of temporal cases. This will help to make airport weather predictions more accurate, which will make air travel safer and make airlines more profitable.

We plan to pursue this research and improve the WIND-1 system in the following ways.

- Test the system with other airports.
- Enable the WIND system to learn autonomously.
- Incorporate additional predictive information, such as user-provided hints, projections of weather radar images of precipitation, and projections of satellite images of cloud.

References

Aamodt, A., and Plaza, E. (1994) Case-based reasoning: Foundational issues, methodological variations, and system approaches, AI Communications, IOS Press, 7, 39–59.

Abarbanel, H. D. I., Brown, R., Sidorowich, J. J., and Tsimring, L. S. (1993) The analysis of observed chaotic data in physical systems, Reviews of Modern Physics, Vol. 65, No. 4, 1331–1367.

Aha, D. W. (1997) The Omnipresence of case-based reasoning in science and application, Technical Report AIC-98-002). Washington, DC: Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence.

Aha, D. W. (1998) Feature weighting for lazy learning algorithms. In Liu, H. and Motoda, H. (Eds.), Feature Extraction, Construction, and Selection: A Data Mining Perspective. Norwell MA: Kluwer, http://www.aic.nrl.navy.mil/papers/1998/AIC-98-003.ps, downloaded September 15, 1999.

Baldwin, J. F., and Martin, T. P. (1995) Fuzzy modelling in an intelligent data browser, Proceedings of the 1995 IEEE International Conference on Fuzzy Systems and the 2nd International Fuzzy Engineering Symposium, IEEE, vol. 5, 1885–1890.

Bardossy, A., Duckstein, L., and Bogardi, I. (1995) Fuzzy rule-based classification of atmospheric circulation patterns, International Journal of Climatology, 15, 1087–1097.

Barry, R. G., and Chorley, R. J., (1968) Atmosphere, Weather and Climate, 4th edition, Methuen & Co. Ltd., New York, New York, USA. 239–240.

Battan, L. J. (1984) Fundamentals of Meteorology, Prentice-Hall, Eaglewood Cliffs, New Jersey, USA, 157–161.

Bezdek, J. C. (1994) Fuzziness vs. probability – Again (!?), IEEE Transactions on Fuzzy Systems, Vol. 2, No. 1, (an editorial article) February 1994, 1–3.

Bezdek, J. C., and Pal, S. K. (eds.) (1992) Fuzzy Models for Pattern Recognition, IEEE Press.

Bonissone, P. P., and Ayub S. (1992) Similarity measures for case-based reasoning systems. In Bouchon_Meunier et al. (eds.) IPMU'92-Advanced Methods in AI: 4th International Conference on Information Processing and Management of Uncertainty in KBS, Palma de Mallorca, July 1992 Proceedings, Spain. Lecture Notes in AI 682, 483–487.

Bonissone, P. P., and Cheetham, W. (1997) Financial applications of fuzzy case-based reasoning to residential property valuation, Proceedings of the 6th IEEE International Conference on Fuzzy Systems. 1997. Vol. 1, 37–44.

Bosc, P. and Pivert, O. (1992) Some approaches for relational databases flexible querying, Journal of Intelligent Information Systems, 1, 323–354.

Bradley, F. M., and Ellis, B. W. (eds.) (1992) Rodale's All-New Encyclopedia of Organic Gardening: The Indispensable Resource For Every Gardener, Rodale Press Inc., Emmaus, Pennsylvania, USA, pgs. 629–630 and 651.

Bridge, D. G. (1998) Defining and combining symmetric and asymmetric similarity measures, Proceedings of the 4th European Workshop On Case-Based Reasoning, 52–63.

Cain, T., Pazzani, M. J., and Silverstein, G. (1991) Using domain knowledge to influence similarity judgement. In Proceedings of the Case-Based Reasoning Workshop, Morgan Kaufmann, Washington, DC, 191–199.

Cats, G., and Wolters, L. (1996) The Hirlam project, IEEE Computational Science and Engineering, Winter 1996, Vol. 3, Issue 4, 4–7. ("Hirlam" is short for High resolution limited area modelling, a co-project of several European countries to develop and maintain a numerical short-range weather forecasting system. Additional information about Hirlam was available on its web site, The international HIRLAM Project, at http://www.knmi.nl/hirlam on September 13, 1999.)

Cheetham, W., and Graf, J. (1997) Case-based reasoning in colour matching, by In, Proc. ICCBR-97, Leake, D. & Plaza, E. (eds.) LNAI, Springer, 1–12.

Christopherson, D. (1998) Artificial intelligence in the weather forecast office-One forecaster's view, Preprints of the 1st Conference on Artificial Intelligence, American Meteorological Society, 136–143.

Clarke, D. A. (1995) Characterizing the causes of low ceiling and visibility at U.S. airports, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 325–330.

Conway, B.J. (1989) Expert systems and weather forecasting, Meteorological Magazine, 118, 23–30.

Cox, E. (1992) The seven truths of fuzzy logic, Computer Design, April 1992.

Creecy, R. H., Masand, B. M., Smith, S. J., and Waltz, D. L. (1992) Trading MIPS and memory for knowledge engineering, Communications of the ACM, 35, 48–64.

Cunningham, P., and Bonzano, A. (1999) Knowledge engineering issues in developing a case-based reasoning application, Knowledge-Based Systems, 12 (1999), 371–379.

Dallavalle, J. P., and Dagostaro, V. J. (1995) The accuracy of ceiling and visibility forecasts produced by the National Weather Service, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 213–218.

Doswell III, C. A., and Brooks, H. E. (1998) Budget cutting and the value of weather services, Weather and Forecasting, Vol. 13, No. 1, 206–212.

Doyle, C. (1995) Calculating the value of forecast availability at aerodromes in Canada: The potential cost of service withdrawal, Evaluation Network Newsletter, Meteorological Service of Canada, Environment Canada, Toronto, Canada. http://www.tor.ec.gc.ca/ppid/evalnet/eng/nl_i5e.htm#calcu, downloaded September 15, 1999.

Dreyfus, H. L. (1992) What Computers Still Can't Do: A Critique of Artificial Reasoning, The MIT Press, Cambridge, Massachusetts, pgs. 126, 128, 322–323.

Dubois, D., Esteva, F., Garcia, P., Godo, L., de Mantaras, L., and Prade, H. (1997) Fuzzy modelling of case-based reasoning and decision, Case-Based Reasoning Research and Development, Proceedings of the 2nd International Conference on Case-Based Reasoning (ICCBR-97); Leake, D. B., and Plaza, E. (eds.), Springer Verlag, Berlin. 599–610.

Dudani, S. A. (1976) The distance-weighted k-nearest neighbor rule, IEEE Transactions on Systems, Man, and Cybernetics, Volume SMC-6, Number 4, April 1976, 325–327.

Dyer, R. M. and Moninger, W. R. (1988) Summary report in the second workshop on artificial intelligence in the environmental sciences (AIRIES), 15-17 September 1987 Boulder Colorado, Bulletin of the American Meteorological Society, 69, 508–514.

Frankel, D. S., Draper, J. S., Peak, J. E., and McLeod, J. C. (1995) Artificial intelligence needs workshop, 4-5 November 1993, Boston, Massachusetts (meeting summary), Bulletin of the American Meteorological Society, Vol. 76, No. 5, 728-738.

Garner, T. (1995) Use of NGM regional profile forecasts for fog forecasting for space shuttle, landing conditions at KSC [Kennedy Space Center], Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 189–191.

Gentner, D. and Forbus, K. (1991) MAC/FAC: A model of similarity-based retrieval, Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society, Chicago, IL: Lawrence Earlbaum, 504–509.

Gershon, N. D., and Miller, C. G. (1993) Dealing with the data deluge (Special report: Environment, part 2), IEEE Spectrum, Vol. 30, No. 7, 28–32.

Gollvik, S., and Olsson, E. (1993) NORTAF: Computer generated aerodrome forecasts, 5th International Conference on Aviation Weather Systems, American Meteorological Society, 335–338.

Göös, J., Ala-Siuru, P., and Korpipää, P. (1999) Fuzzy monitoring and case-based diagnosis tool for electronics manufacturing process, Proceedings of the 4th UK Case-Based Reasoning Workshop (UKCBR4), University of Salford, UK. (A copy of this paper was kindly given to me by Janne Göös.)

Hadjimichael, M., Kuciauskas, A. P., Brody, L. R., Bankert, R. L., and Tag, P. M. (1996) MEDEX: A fuzzy system for forecasting Mediterranean gale force winds, Proceedings of FUZZ-IEEE 1996 IEEE International Conference on Fuzzy Systems, 529–534.

Hansen, B. K. (1997) SIGMAR: A fuzzy expert system for critiquing marine forecasts, AI Applications, Vol. 11, No. 1, 59–68.

Hansen, B. K., and Riordan, D. (1998) Fuzzy case-based prediction of ceiling and visibility, Preprints of the 1st Conference on Artificial Intelligence, American Meteorological Society, 118–123.

Hansen, B. K., Bankert, R., and Hadjimichael, M. (eds.) (1999) Fuzzy logic in environmental sciences: A bibliography, http://chebucto.ns.ca/~bjarne/fuzzy_environment, downloaded Sept. 27, 1999.

Hashemi, S. (1999) A review of case-based reasoning, A report for a doctoral directed studies course at the Faculty of Computer Science, DalTech, Dalhousie University, Halifax, Canada (submitted to Professor Denis Riordan).

Huschke, R. E. (editor) (1959) Glossary of Meteorology, American Meteorological Society, Boston, Massachusetts, USA, pgs. 106 and 419.

Jeng, B. C., and Liang, T.-P. (1995) Fuzzy indexing and retrieval in case-based systems, Expert Systems With Applications, Vol. 8., No. 1, 1995. Elsevier Science Ltd., 135–142.

Jones, E., and Roydhouse, A. (1995) Intelligent retrieval of archived meteorological data, IEEE Expert, Vol. 10, No. 6, 50–57.

Kamp, G., Lange, S., and Globig, C. (1998) Related Areas [to CBR. Final chapter]. In Lenz, M., Bartsch-Spörl, B., Burkhard, H.-D., and Wess, S. (eds.), Case-based reasoning technology) From foundations to applications, Springer-Verlag, Berlin Heidelberg, 325–351.

Kantrowitz, M., Horstkotte, E., Joslyn, C. (eds.) (1997) Answers to Questions about Fuzzy Logic and Fuzzy Expert System (Fuzzy Logic FAQ, i.e., Frequently Asked Questions), http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-9.html, downloaded September 11, 1999.

Kasriel, R. H. (1971) Undergraduate Topology, W. B. Saunders Company, Philadelphia, PA, U.S.A., pgs. 81–82.

Keller, J. M., Gray, M. R., and Givens Jr., J. A. (1985) A fuzzy k-nearest neighbor algorithm, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 15, No. 4, 258–263.

Keller, J., Smith, C., and Wilson, F. W. (1995) Applications of column models for terminal weather nowcasts, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 66–71.

Kilpinen, J. (1993) The combination of forecasts in an automated aviation weather forecasting system, 5th International Conference on Aviation Weather Systems, American Meteorological Society, 339–340.

Kolodner, J. (1993) Case-Based Reasoning, Morgan Kaufmann, San Francisco, California, USA.

Kontkanen, P., Myllymäki, P., Silander, T., and Tirri, H. (1998) On Bayesian case matching, Advances in case-based reasoning: Proceedings of the Fourth European Workshop on CBR/EWCBR-98, 13–23.

Kosko, B. (1997) Fuzzy Engineering, Prentice Hall, New Jersey, USA.

Kuciauskas, A. P., Brody, L. R., Hadjimichael, M., Bankert, R. L., and Tag, P. M. (1998) MEDEX: Applying fuzzy logic to a meteorological expert system, Preprints of the 1st Conference on Artificial Intelligence, American Meteorological Society, 68-74.

Kumar, V. R., Chung, C. Y. C., and Lindley, C. A. (1994) Toward building and expert system for weather forecasting operations, Expert Systems with Applications, Vol. 7, No. 2, Permagon, Elsevier Science Ltd., 373–381.

Langly, P. (1996) Elements of Machine Learning, Morgen Kaufmann Publishers, San Francisco, California, USA., pp. 381–382.

Leake, D. B. (1996) CBR in context. The present and future; in Leake, D. B. (editor) (1996) Case-Based Reasoning: Experiences, Lessons & Future Directions, American Association for Artificial Intelligence, Menlo Park California, USA, 3–30.

Lefley, M., and Austin, V. (1997) Match'em: Using fuzzy logic to profile criminals. Proceedings of 6th. IEEE conference on fuzzy logic '97 July 1997, Barcelona, Spain.

Leigh, R. J. (1995) Economic benefits of Terminal Aerodrome Forecasts (TAFs) for Sydney Airport, Australia, Meteorological Applications, 2, 239–247.

Liao T. W., Zhang Z., and Mount, C. R. (1998) Similarity measures for retrieval in case-based reasoning systems, Applied Artificial Intelligence, 12, 267–288.

López de Mántaras, R., and Plaza, E. (1997) Case-based reasoning: An overview, AI Communications, 10, 21–29.

Lorenz, E. N. (1963) Deterministic nonperiodic flow, Journal of the Atmospheric Sciences, 20, 130–141.

Lorenz, E. N. (1969a) Three approaches to atmospheric predictability, Bulletin of the American Meteorological Society, 50, 345–349.

Lorenz, E. N. (1969b) Atmospheric predictability as revealed by naturally occurring analogues, Journal of the Atmospheric Sciences, 26, 636–646.

Lorenz, E. N. (1977) An experiment in nonlinear statistical weather forecasting, Monthly Weather Review, 105, 590–602.

Lorenz, E. N. (1993) The Essence of Chaos, University of Washington Press, Seattle, WA, USA.

Luger, G. F., and Stubblefield, W. A. (1998) Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Addison Wesley Longman, Reading, Massachusetts, USA, pg. 238.

Macdonald, K. (1999) Personal correspondence. April 1 1999, Ken Macdonald is Acting Director of the Inter-Agency Services Branch, AES, Environment Canada.

Macdonald, K. (1998) Personal correspondence. July 10, 1998. Ken is Chief of Client Services and Market Management for the Interagency Services Branch of Environment Canada.

Main. J., Dillon, T. S., and Khosla. R. (1996) Use of fuzzy feature vectors and neural vectors for case retrieval in case based systems, NAFIPS 1996 Biennial Conference of the North American Fuzzy Information Processing Society, IEEE, New York, NY, 438–443.

Maner, W., and Joyce, S. (1997) WXSYS: Weather Lore + Fuzzy Logic = Weather Forecasts, presented at 1997 CLIPS Virtual Conference, http://web.cs.bgsu.edu/maner/wxsys/wxsys.htm, downloaded Feb. 27, 1999. (Maner offers the following citation information: "WXSYS," published on the CD-ROM (ISBN 0-534-95053-1) that accompanies Giarratano and Riley, Expert Systems: Principles and Programming, third edition (Boston: PWS Publishing Company, 1998). This program was one of only 6 programs selected for publication after an international competition conducted in 1997 as the CLIPS Virtual Conference. This artificial intelligence program, written in FuzzyCLIPS, may be the first fuzzy-logic expert system to predict local weather based on a compendium of common-sense rules extracted from weather lore. The program was 8673 lines as published, about half code and half documentation, and used 95 fuzzy prediction rules. The publisher required demonstration and validation code, which occupies an additional 3829 lines on the CD-ROM. With Sean Joyce. This paper was one of only 17 papers selected for publication after an international competition conducted in 1997 as the "CLIPS Virtual Conference". Joseph Giarratano and PWS Publishing Company hosted the conference. With Sean Joyce. Fifteen journal pages, PDF format.)

Martin, D. E. (1972) Climatic presentations for short-range forecasting based on event occurrence and reoccurrence profiles, Journal of Applied Meteorology, 11, 1212–1223.

McBratney, A. B., and Moore, A. W. (1985), Application of fuzzy sets to climatic classification, Agricultural and Forest Meteorology, 35, 165–185.

Meadows, N. (1997) TAF tools project-Phase one report, [specially commissioned for the] Interagency Services Report, National Weather Services Directorate, Meteorological Service of Canada, Environment Canada.

Meyer, W. D. (1995) A hybrid model for predicting fog and stratus clouds, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 441–444.

Mosher, F. R. (1998) Meteorological craftsman vs. meteorological engineers, Preprints of the 1st Conference on Artificial Intelligence, American Meteorological Society, 164–167.

Munataka, T., and Jani, Y. (1994) Fuzzy systems: An overview, Communications of the ACM, Vol. 37, No. 3, 69–76.

Murtha, J. (1995) Applications of fuzzy logic in operational meteorology, Scientific Services and Professional Development Newsletter, Canadian Forces Weather Service, 42–54.

Patton, R. (1996) Analysis of user requirement for commercial/transport aviation, Forecasting Research Technical Report No. 204, Meteorological Office, Bracknell, Berkshire, United Kingdom.

Pearson, W. R., and Lipman, D. J. (1988) Improved tools for biological sequence comparison, Proceedings of the National Academy of Sciences, Vol. 85, 2444–2448, April, 1988, Biochemistry.

Porter, C. W., and Seaman, N. L. (1995) Short-term high resolution forecasting of cloud ceiling and visibilities, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 331–336.

Rafiei, D. (1999) Fourier-Transform Based Techniques in Efficient Retrieval of Similar Time Sequences, Ph.D. thesis, Department of Computer Science, University of Toronto, Ontario, Canada.

Riesbeck, C. K. (1996) What next? The future of CBR in post-modern AI. In Case-Based Reasoning: Experiences, Lessons & Future Directions, Leake, D. B. (editor), American Association for Artificial Intelligence, Menlo Park California, USA, 371–388.

Riesbeck, C. K., and Schank, R. C. (1989) Inside Case-Based Reasoning, Lawrence Erlbaum Associates, Hillside, New Jersey, USA.

Rougegrez, S. (1993) Similarity evaluation between observed behaviors and the prediction of processes. In Wess, S., Althoff, K. D. and Richter, M. (eds.), Topics in Case-Based Reasoning, Proceedings First European Workshop on Case-Based Reasoning, 1993, Springer-Verlag, Berlin, 155–166.

Shakina, N. P., Ivanova, A. R., Sheveleva, O. V., and Kalugina, G. Yu. (1993) Aviation weather forecasting automated methods in the RAFC Moscow and the Airport Vnukovo, 5th International Conference on Aviation Weather Systems, American Meteorological Society, 341–342.

Shivakumar, N., and Garcia-Molina, H. (1995) SCAM: A copy detection mechanism for digital documents, Digital Libraries ‘95, The Second Annual Conference on the Theory and Practice of Digital Libraries, 155–163.

Silvert, W. (1981) The formulation and evaluation of predictions, International Journal of General Systems, Vol. 7, 189–205.

Singh, S. (1999) A single nearest-neighbor fuzzy approach for pattern recognition, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 13, No. 1, 49–64.

Sivillo, J. K., Ahlquist, J. E., and Toth, Z. (1997) An Ensemble Forecasting Primer, Weather and Forecasting, Vol. 12, No. 4, 809–818.

Srinivasan, D., Liew, A. C., and Chang, C. S. (1994) Forecasting daily load curves using a hybrid fuzzy-neural approach, IEE Proceedings on Generation, Transmission and Distribution, Vol. 141, No. 6, 561–567.

Stanski, H. (1999) Personal correspondence. The commonest cause for TAFs needing to be amended is the occurrence of unforecast categories of cloud ceiling and visibility.

Stanski, H., Leganchuk, A., Hanssen, A., Wintjes, D., Abramowski, O., and Shaykewich, J. (1999) NAV CANADA's TAF amendment response time verification , Eighth Conference on Aviation, Range, and Aerospace Meteorology, 10-15 January 1999, Dallas, Texas, American Meteorological Society, 63–67.

Stern, H., and Parkyn, K. (1999) Predicting the likelihood of fog at Melbourne Airport, Eighth Conference on Aviation, Range, and Aerospace Meteorology, 10-15 January 1999, Dallas, Texas, American Meteorological Society, 174–178.

Sujitjorn, S., Sookjaras, P., Wainikorn, W. (1994) An expert system to forecast visibility in Don-Muang Air Force Base, 1994 IEEE International Conference on Systems, Man and Cybernetics (Humans, Information and Technology) (2-5 Oct., 1994), IEEE, NY, NY, USA, 2528–2531.

Tag, P. M., Hadjimichael, M., Brody, L. R., Kuciauskas, A. P., and Bankert, R. L. (1996) Automating the subjective recognition of 500 mb wind patterns as input to a meteorological forecasting system, Proceedings of the 15th Conference on Weather Analysis and Forecasting, American Meteorological Society, 347-350.

Tobin, K. W., Gleason, S. S., and Karnowski, T. P. (1998) Adaptation of the fuzzy k-nearest neighbor classifier for manufacturing automation, International Society for Optical Engineering Photonics, West Symposium on Electronic Imaging, San Jose, CA. January 22-30, 1998 (http://www-ismv.ic.ornl.gov/publications/kwt_spie98.pdf, downloaded March 6, 2000)

US National Weather Service Aviation Weather Center, URL http://www.awc-kc.noaa.gov/awc/help/what_is_a_taf.html, downloaded Jan. 29, 1999.

Van den Dool, H. M. (1989) A new look at weather forecasting through analogues, Monthly Weather Review, 117, 2230–2247.

Van den Dool, H. M. (1994) Searching for analogues, how long must we wait?, Tellus, 45A, 314–324.

Viot, G. (1993) Fuzzy logic in C, Dr. Dobb's Journal, February 1993 (special issue on "cognitive computing"), pgs. 40–49 and 94.

Vislocky, R. L., and Fritsch, J. M. (1997) An automated, observations-based system for short-term prediction of ceiling and visibility, Weather and Forecasting, 12, 31–43.

Warner, T. T., and Stoelinga (1995) Very-very-short-range numerical simulation of ceiling and visibility during an East Coast snowstorm, Preprints of the 6th Conference on Aviation Weather Systems, American Meteorological Society, 141–143.

Watson, I., and Marir, F. (1994) Case-based reasoning: A review, Knowledge Engineering Review, Vol. 9, No. 4. (http://www.ai-cbr.org/classroom/cbr-review.htm, downloaded November 13, 1998)

Weber-Lee, R., Barcia, R. M., and Khator, S. K. (1995) Case-based reasoning for cash flow forecasting using fuzzy retrieval. Case-Based Reasoning Research and Development; Veloso, M., and Aamodt, A. (eds.), First International Conference on Case-Based Reasoning (ICCBR-95), Sesimbra, Portugal, Springer, 1995.

Whiffen, B. (1993) FTGEN - An automated FT production system, 5th International Conference on Aviation Weather Systems, American Meteorological Society, 327–330.

White, J. F. (1995) Aeronautical meteorology and the International Air Transport Association (IATA), World Meteorological Organization Bulletin, 44 (4), 335–339.

White, P.W. (1987) Impact of weather forecasts on aviation fuel consumption, The Meteorological Magazine, No. 1375, Vol. 115, pg. 29.

Wilson, L. J., and Sarrazin, R. (1989) A classical-REEP [Regression Estimation of Event Probabilities] short-range forecast procedure, Weather and Forecasting, 4, 502–516.

Winder, L., Lefley, M., and Smith, B. (1997) A key for freshwater invertebrates using fuzzy logic. Computer Applications in the Biological Sciences, February 1997.

Xia, B. B. (1997) Similarity Search in Time Series Data Sets, Master of Science thesis, Department of Computer Science, Simon Fraser University, BC, Canada.

Yager, R. R. (1997) Case-based reasoning, fuzzy systems modelling and solutions composition, Case-Based Reasoning Research and Development, Proceedings of the 2nd International Conference on Case-Based Reasoning (ICCBR-97); Leake, D. B., and Plaza, E. (eds.), Springer Verlag, Berlin. 633–642.

Yang, M. S. (1993) A survey of fuzzy clustering, Math. Comp. Modelling, vol. 18, no 11, 1–16.

Zadeh, L. A. (1965) Fuzzy sets, Information and Control, Vol. 8 (June 1965), 338-353; reprinted in Bezdek, J. C., and Pal, S. K. (eds.) (1992) Fuzzy Models for Pattern Recognition, IEEE Press, 35–45.

Zadeh, L. A. (1999) From computing with words to computing with perceptions—from manipulation of measurements to manipulation of perceptions, IEEE Transactions on Circuits and Systems—1: Fundamental Theory and Applications, Vol. 45, No. 1, 105–119.

Zimmerman, H. J. (1991) Fuzzy Set Theory and its Applications (2nd edition), Kluwer Academic Publishers.

Additional References on Analog Forecasting in Meteorology

Here is a list of meteorology papers that describe the method of analog forecasting. All of these papers contributed to our understanding but only a few of these papers are cited specifically in this thesis.

Barnett, T. P., and Preisendorfer, R. W. (1978) Multifield analog prediction of short-term climate fluctuations using a climate state vector, Journal of Atmospheric Science, 35, 1771–1787.

Barnston, A. G., and Livezey, R. E. (1989) An operational multifield analog / anti-analog prediction system for United States seasonal temperatures. Part II: Spring, summer, fall and intermediate 3-month period experiments, Journal of Climate, 2, 513–541.

Bergen, R.E., and Harnack, R. P. (1982) Long-range temperature prediction using a simple analog approach, Monthly Weather Review, 110, 1083–1099.

Guilbaud, S. (1997) *Prevision Quantitative des Precipitations Jounalieres par une Method Statistico-Dynamique de Recherche d'Analogues* [Prediction of Quantitative Precipitation by a Statistical-Dynamical Method for Finding Analogs], Ph.D Thesis, L'Institute National Polytechnique de Grenoble, France.

Huschke, R. E. (editor) (1959) Glossary of Meteorology, American Meteorological Society, Boston, Massachusetts, USA, pgs. 106 and 419.

Kruizinga, S., and Murphy, A. H. (1983) Use of an analogue procedure to formulate objectively probabilistic temperature forecasts in the Netherlands, Monthly Weather Review, 111, 2244–2254.

Livezey, R. E., and Barnston, A. G. (1988) An operational multifield analog/antianalog system for United States seasonal temperatures. Part 1: System design and winter experiments, Journal of Geophysical Research, Vol. 93, No. D9, 10953–10974.

Lorenz, E. N. (1963) Deterministic nonperiodic flow, Journal of the Atmospheric Sciences, 20, 130–141.

Lorenz, E. N. (1969a:Three approaches to atmospheric predictability, Bulletin of the American Meteorological Society, 50, 345–349.

Lorenz, E. N. (1969b) Atmospheric predictability as revealed by naturally occurring analogues, Journal of the Atmospheric Sciences, 26, 636–646.

Lorenz, E. N. (1977) An experiment in nonlinear statistical weather forecasting, Monthly Weather Review, 105, 590–602.

Lorenz, E.N. (1993) The Essence of Chaos, University of Washington Press, Seattle, WA, USA.

Martin, D. E. (1972) Climatic presentations for short-range forecasting based on event occurrence and reoccurrence profiles, Journal of Applied Meteorology, 11, 1212–1223.

Nicolis, C. (1998) Atmospheric Analogs and Recurrence Time Statistics: Toward a Dynamical Formulation. Journal of the Atmospheric Sciences, Vol. 55, No. 3, 465–475.

Radinivic, D. (1975) An analogue method for weather forecasting using the 500/1000 mb relative topography, Monthly Weather Review, 103, 639–649.

Roebber, P. J., Bosart, L. F. (1998) The Sensitivity of Precipitation to Circulation Details. Part I: An Analysis of Regional Analogs, Monthly Weather Review, Vol. 126, No. 2, 437–455.

Ruosteenoja, K. (1988) Factors affecting the occurrence and lifetime of 500 mb height analogues: A study based on a large amount of data, Monthly Weather Review, 116, 368–376.

Soucy, D. (1991) Revised users guide to days 3-4-5 automated forecast composition program, CMC Technical Document No. 37, Canadian Meteorological Centre, Environment Canada.

Toth, Z. (1989) Long-range weather forecasting using an analog approach, Journal of Climate, 2, 594–607.

Toth, Z. (1991a) Intercomparison of circulation similarity measures, Monthly Weather Review, 119, 55–64.

Toth, Z. (1991b) Estimation of the atmospheric predictability by circulation analogs, Monthly Weather Review, 119, 65–72.

Toth, Z. (1991c) Circulation patterns in phase space: A multinormal distribution?, Monthly Weather Review, 119, 1501–1511.

Van den Dool, H. M. (1994) Searching for analogues, how long must we wait?, Tellus, 45A, 314–324.

Van den Dool, H. M. (1989) A new look at weather forecasting through analogues, Monthly Weather Review, 117, 2230–2247.

Van den Dool, H. M. (1987) A bias skill in forecasts based on analogues and antilogues, Journal of Applied Meteorology, 26, 1278–1281.

Van den Dool, H. M. (1989) A new look at weather forecasting through analogues, Monthly Weather Review, 117, 2230–2247.

Woodcock, F. (1980) On the use of analogues to improve regression forecasts, Monthly Weather Review, 108, 292–297.

Appendix A: Sample Questionnaire for Knowledge Acquisition

Knowledge acquisition is performed simply by having the expert fill in a questionnaire such as the one shown below. When such a questionnaire is completed, it contains all the information needed to construct the fuzzy sets and to order the fuzzy operations that shown above in Section 3.2 (pg. 78). The grayed-out fields would normally be blank but we have inserted sample values for the *WIND-1* configuration.

Part A: Attributes and order of comparison

Specify the attributes to compare and the order in which they are to be compared.

date of the year, hour of the day, cloud amount, cloud ceiling height, visibility, wind direction, wind speed, precipitation type, precipitation intensity, dew point temperature, dry bulb temperature, pressure trend

Part B: Continuous-number attributes

List the continuous-number attributes—those with values that can be positive or negative and which are compared in terms of their relative difference—and for each, specify values of difference that signify *slightly near*, *near*, and *very near*. If you choose to fill in only the middle column, then, by default, the threshold for *slightly near* will be twice that for *near*, and the threshold for *very near* will be half that for *near*.

Attribute	***slightly near***	***near***	***very near***
date of the year	*60 days*	*30 days*	*10 days*
hour of the day	*2 hours*	*1 hours*	*0.5 hours*
wind direction	*40 degrees*	*20 degrees*	*10 degrees*
dew point temperature	*4 degrees*	*2 degrees*	*1 degree*
dry bulb temperature	*8 degrees*	*4 degrees*	*2 degree*
pressure trend	$0.20\ kPa \cdot hr^{-1}$	$0.10\ kPa \cdot hr^{-1}$	$0.05\ kPa \cdot hr^{-1}$

Part C: Absolute-number attributes

List the absolute-number attributes—those with values that can be only equal to zero or positive numbers, and which are compared in terms of their relative magnitudes—and for each possible pair, use numbers in the range (0.0…1.0] to specify how near they are to each other. The number 0.25 corresponds to *slightly near*, the number 0.50 corresponds to *near*, and the number 0.75 corresponds to *very near*.

Attribute

wind speed

0	1.00					
1	*0.75*	1.00				
2	*0.50*	*0.75*	1.00			
3	*0.25*	*0.50*	*0.75*	1.00		
4	*0.10*	*0.25*	*0.50*	*0.75*	1.00	
…	…	…	…	…	…	…
	0	**1**	**2**	**3**	**4**	**…**

cloud amount **…**

cloud ceiling height **…**

visibility **…**

Part D: Nominal attributes

List the nominal attributes, and for each possible pair, use numbers in the range (0.0…1.0] to specify how near they are to each other The number 0.25 corresponds to *slightly near*, the number 0.50 corresponds to *near*, and the number 0.75 corresponds to *very near*. [68]

Attribute

precipitation type	*Nil*	1.00				
	Drizzle	*0.02*	1.00			
	Showers	*0.03*	*0.50*	1.00		
	Rain	*0.01*	*0.50*	*0.75*	1.00	
	…	…	…	…	…	
		Nil	Drizzle	Showers	Rain	…

precipitation intensity	…

[68] The nominal attribute table shows the "distance" between any two nominal attributes in the same way that a distance table on a highway map shows distances between towns. For illustration purposes, the table describes fuzzy relationships between only four common precipitation types. The actual WIND-1 configuration table specifies fuzzy relationships between 24 possible precipitation types (e.g, freezing rain, ice pellets, etc.).

Part E: Recency

For each possible time step in comparable temporal cases, use numbers in the range (0.0…1.0] to specify the lowest level of similarity that can be attributed to attributes from that time step. The number 0.25 corresponds to *slightly near*, the number 0.50 corresponds to *near*, and the number 0.75 corresponds to *very near*.

Time in case	**Minimum similarity**	**Time in case**	**Minimum similarity**
…	…	t-0	*0.00*
t-12	*0.96*	t+1	*0.00*
t-11	*0.95*	t+2	*0.00*
t-10	*0.94*	t+3	*0.00*
t-9	*0.93*	t+4	*0.00*
t-8	*0.92*	t+5	*0.00*
t-7	*0.91*	t+6	*0.00*
t-6	*0.90*	t+7	*0.00*
t-5	*0.80*	t+8	*0.00*
t-4	*0.70*	t+9	*0.00*
t-3	*0.60*	t+10	*0.00*
t-2	*0.40*	t+11	*0.00*
t-1	*0.20*	t+12	*0.00*
		…	…

Appendix B: A Worked-out Example of Fuzzy *k*-nn Algorithm for Prediction

We hope that the results achieved by the fuzzy *k*-nn algorithm are reproducible, in weather prediction and in other applications. Towards that end, this appendix presents a step-by-step, worked-out example of the fuzzy *k*-nn algorithm described in Section 3.2 (pg. 78).

We begin by assuming the similarity-measuring function has been configured as explained in Section 3.2 and in Appendix A. The three main processes in using the algorithm are as follows.

1. Measure similarity of temporal cases.
2. Traverse case base to find *k*-nn.
3. Make prediction based on weighted median of *k*-nn.

The first process, is the most original, so this appendix presents a detailed example of how similarity of cases is calculated. The following example pertains to weather but it technique ought to generalize to any sort of application which describes temporal cases in terms of continuous, absolute, and nominal attributes.

Measure similarity

A present case is composed of actual weather observations are shown in Figure 33.

```
CYHZ 120000Z 19010G16KT 1 1/2SM SHRA BR OVC007 21/20 A2990 RMK SF8 RERA SLP125=
CYHZ 120100Z 20011KT 1 1/2SM -SHRA BR BKN003 OVC007 21/20 A2991 RMK SF5SF3 SLP127=
CYHZ 120046Z 20012KT 3SM -SHRA BR OVC007 RMK SF8 RERA=
CYHZ 120200Z 23011KT 8SM BKN004 OVC011 21/20 A2992 RMK SF5SC3 SLP130=
CYHZ 120249Z 23008KT 2SM BR OVC007 RMK SF8 CIG RGD=
CYHZ 120300Z 23008KT 2SM BR OVC005 21/20 A2994 RMK SF8 SLP139=
CYHZ 120400Z 21009KT 10SM FEW007 OVC076 20/19 A2993 RMK SF2AC6 SLP135=
CYHZ 120500Z 32006KT 10SM BKN010 OVC075 19/18 A2995 RMK SC5AC3 SLP140=
CYHZ 120600Z 32005KT 12SM BKN008 OVC210 19/18 A2996 RMK SC6CI2 SLP143=
CYHZ 120700Z 33005KT 15SM FEW008 BKN250 18/17 A2997 RMK SF1CI1 SLP148=
CYHZ 120800Z 33006KT 15SM VCFG SKC 17/16 A2997 RMK VSBY NW 1/2 SLP149=
CYHZ 120900Z 31007KT 10SM PRFG FEW100 BKN250 16/16 A2998 RMK AC1CI1 VSBY LWR W SLP152=
CYHZ 121000Z 29005KT 12SM BKN250 16/15 A3001 RMK CI5 SLP161=
CYHZ 121100Z 29009KT 12SM BKN250 16/15 A3002 RMK CI5 SLP166=
CYHZ 121200Z 29006KT 15SM BKN250 16/14 A3003 RMK CI5 SLP170=
```

Figure 33. Actual weather observations (METAR code) for Halifax International Airport for the period 00:00 to 12:00 UTC 12 September 1999 (obtained from the *Texas A&M Weather Interface* website, http://www.met.tamu.edu/personnel/students/weather/weather_interface.html, downloaded September 12, 1999).

Three simplified weather cases are shown in Figure 34. Case 1 represents the present case to predict for; it is drawn from the data above in Figure 33. Cases 2 and 3 represent two analogs from the weather archive to make predictions from; they are hypothetical. For purposes of illustration, only seven-hour-long cases are considered and only three weather attributes are presented: cloud ceiling, wind direction, and precipitation type; these attributes are, respectively, absolute, continuous, and nominal (as described in Section 3.2, pg. 78). Longer cases with more attributes would be handled by straightforward extension of the technique shown.

time	case 1 cloud ceiling (dam)	case 1 wind dirn. (deg.)	case 1 pcpn.	case 2 cloud ceiling (dam)	case 2 wind dirn. (deg.)	case 2 pcpn.	case 3 cloud ceiling (dam)	case 3 wind dirn. (deg.)	case 3 pcpn.
t-3	9	200	shwrs	12	190	rain	9	170	drzl
t-2	12	230	nil	15	220	nil	9	210	nil
t-1	15	230	nil	21	220	nil	12	220	nil
t-0	21	210	nil	30	220	nil	15	210	nil
t+1	(30)	320	nil	24	330	nil	21	310	nil
t+2	(24)	320	nil	30	330	nil	24	310	nil
t+3	(999)	330	nil	999	340	nil	750	320	nil

Figure 34. Present case (1) and two analogs (2 and 3). Present case is weather from Halifax International Airport for the period 01:00 to 07:00 UTC 14 September 1999. Analogs are contrived for illustration purposes. The *t*-0 observation corresponds to a forecast start time of 04:00 UTC. In a forecast setting, the grayed-out observations in case 1 are not known, however auxiliary predictors (guidance) for the values of wind direction and precipitation are commonly available.

The three attributes presented Figure 34 are sufficient to demonstrate each of fuzzy similarity-measuring operations described in Section 3.2, namely μ^a, μ^c, μ^n, and $\mu^f(t)$ as shown in Figure 35 and Figure 36.

time	ceiling			wind dirn.			pcpn.		
	case 1	case 2	μ^a_2	case 1	case 2	μ^c_2	case 1	case 2	μ^n_2
t-3	9	12	0.75	200	190	0.88	shwrs	rain	0.75
t-2	12	15	0.80	230	220	0.88	nil	nil	1.00
t-1	15	21	0.71	230	220	0.88	nil	nil	1.00
t-0	21	30	0.70	210	220	0.88	nil	nil	1.00
t+1	?	240	-	320	330	0.88	nil	nil	1.00
t+2	?	300	-	320	330	0.88	nil	nil	1.00
t+3	?	999	-	330	340	0.88	nil	nil	1.00

(a) Comparing case 2 to case 1, μ^a_2 is the similarity between their absolute values of ceiling height, μ^c_2 is the similarity between their continuous values of wind direction and μ^n_2 is the similarity between their nominal types of precipitation.

time	ceiling			wind dirn.			pcpn.		
	case 1	case 3	μ^a_3	case 1	case 3	μ^c_3	case 1	case 3	μ^n_3
t-3	9	9	1.00	200	170	0.38	shwrs	drzl	0.50
t-2	12	9	0.75	230	210	0.50	nil	nil	1.00
t-1	15	12	0.80	230	220	0.88	nil	nil	1.00
t-0	21	15	0.71	210	210	1.00	nil	nil	1.00
t+1	?	21	-	320	310	0.88	nil	nil	1.00
t+2	?	24	-	320	310	0.88	nil	nil	1.00
t+3	?	750	-	330	320	0.88	nil	nil	1.00

(b) Comparing case 3 to case 1, μ^a_3 is the similarity between their absolute values of ceiling height, μ^c_3 is the similarity between their continuous values of wind direction, and μ^n_3 is the similarity between their nominal types of precipitation.

Figure 35. Similarity measurement between a present case (1) and two past cases (2 and 3). The fuzzy operations μ^a, μ^c, and μ^n are described in Section 3.2 (pg. 78).

The grayed-out values of wind direction and precipitation for the future parts of the present case (case 1) in Figure 35 are prevision obtained from auxiliary predictors, such as computer models or humans. As explained earlier, existing methods forecast large-scale phenomena, such as wind and precipitation, more effectively than they forecast small-scale phenomena, such as cloud ceilings at a particular airport.

t	ceiling μ^a_2	$\mu^f(t)$	max^a_2	wind dirn. μ^c_2	$\mu^f(t)$	max^c_2	pcpn. μ^n_2	$\mu^f(t)$	max^n_2
-3	0.75	0.60	0.75	0.88	0.60	0.88	0.75	0.6	0.75
-2	0.80	0.40	0.80	0.88	0.40	0.88	1.00	0.4	1.00
-1	0.71	0.20	0.71	0.88	0.20	0.88	1.00	0.2	1.00
-0	0.70	0.00	0.70	0.88	0.00	0.88	1.00	0.0	1.00
1				0.88	0.00	0.88	1.00	0.0	1.00
2				0.88	0.00	0.88	1.00	0.0	1.00
3				0.88	0.00	0.88	1.00	0.0	1.00
		min =	**0.70**		*min =*	**0.88**		*min =*	**0.75**

$min\ \{ max^a_2, max^c_2, max^n_2 \} = min\ \{$ 0.70, 0.88, 0.75 $\} =$ **0.70**

(a) For case 2, the "min of the maxes" equals 0.70, due to a dissimilarity between cloud ceilings at time *t-0*. Assign to case 2 this value of similarity to case 1.

t	ceiling μ^a_3	$\mu^f(t)$	max^a_3	wind dirn. μ^c_3	$\mu^f(t)$	max^c_3	pcpn. μ^n_3	$\mu^f(t)$	max^n_3
-3	1.00	0.60	1.00	0.38	0.60	0.60	0.50	0.6	0.60
-2	0.75	0.40	0.75	0.50	0.40	0.50	1.00	0.4	1.00
-1	0.80	0.20	0.80	0.88	0.20	0.88	1.00	0.2	1.00
-0	0.71	0.00	0.71	1.00	0.00	1.00	1.00	0.0	1.00
1				0.88	0.00	0.88	1.00	0.0	1.00
2				0.88	0.00	0.88	1.00	0.0	1.00
3				0.88	0.00	0.88	1.00	0.0	1.00
		min =	**0.71**		*min =*	**0.50**		*min =*	**0.60**

$min\ \{ max^a_3, max^c_3, max^n_3 \} = min\ \{$ 0.71, 0.50, 0.60 $\} =$ **0.50**

(b) For case 3, the "min of the maxes" equals 0.50, due to a dissimilarity between their wind directions at time *t-2*. Assign to case 3 this value of similarity to case 1.

Figure 36. Raise old low values of similarity—in effect, "forget" old dissimilarities with $\mu^f(t)$. Then determine the minimum of the maximum of all the similarities between past case and present case. The fuzzy operation μ^f is described in Section 3.2 (pg. 78).

The just described process of similarity measurement of temporal cases is the most complicated process in the fuzzy *k*-nn algorithm for prediction. The subsequent two process are relatively simple and explained briefly as follows.

Traverse case base

Traverse the case base measuring the similarity between past cases and a present case and simultaneously maintain a linked list of the *k* most similar cases (such as is shown in Figure 18 on pg. 88). Make every case-to-case similarity measuring process only as detailed as necessary. If initial attribute-to-attribute tests imply strong dissimilarity between cases—sufficient to exclude the past case in question form the *k*-nn set—then terminate the similarity measurement process for that past case and proceed to the next past case.

Make predictions based on a weighted median of the *k*-nn

For purposes of illustration, we assume that we sought only two analogs for the present case, that is, $k = 2$. The weighted median calculation easily extends to higher values of *k*.

Figure 36 shows that between case 2 and case 1 the degree of similarity equals 0.70, and between case 3 and case 1 the degree of similarity equals 0.50. Hence, a prediction for case 1 should consist of such proportional parts of case 2 and case 3, as shown in Figure 37.

time	case 2	case 3		prediction	(actual)
t+1	(0.7 * **24**	+ 0.5 * **21**)	/ (.7+.5) =	**23**	**(30)**
t+2	(0.7 * **30**	+ 0.5 * **24**)	/ (.7+.5) =	**28**	**(24)**
t+3	(0.7 * **999**	+ 0.5 * **750**)	/ (.7+.5) =	**895**	**(999)**

Figure 37. Prediction based on weighted median of *k*-nn ($k = 2$).