MODELLING READERS OF NEWS ARTICLES USING NESTED BELIEFS

Christine Gerard

A thesis
in
The Department
of
Computer Science

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Computer Science
Concordia University
Montréal, Québec, Canada

August 2000

0-612-54331-5

Abstract

Modelling Readers of News Articles Using Nested Beliefs

Christine Gerard

Due to the wealth of documents available on-line, information retrieval on the Internet, in document databases or newspaper web sites can often lead to hundreds of documents being selected of which only a small number are of any interest to the user. This thesis is concerned with creating models of readers of news articles which could be used to filter and evaluate information to present to the reader only those articles containing information relevant to the reader's search.

News articles are of special interest since they consist of information told to a reporter by some sources. Such reported speech can only be adequately modelled using nested belief models to represent the reader's beliefs about the reporter's beliefs about the source's beliefs. This thesis proposes a method for representing a news article, analysing it to extract encoded information, determining the reliability of the information reported in the article, creating models of all the agents in the article, and simulating how a reader of news articles acquires or adopts information from the sources in the article.

A system called *Percolator* is presented. *Percolator* is a stand-alone implementation of one component of the more complex system required to model the nested beliefs of readers of news articles. It demonstrates how a technique called modified belief percolation can be used to simulate how a reader can acquire beliefs from the sources in an article.

Acknowledgments

I wish to acknowledge the support given to me by my family, my friends, and my colleagues throughout my years as a master's student and especially in the last and most critical months.

I dedicate this thesis to Mahmoud without whose not so subtle prodding I never would have finished this difficult task and, to my sons, Alexandre and Jason, for putting up with my long hours of working at the computer when what they really wanted was for me to play with them.

A special thank you to my parents for instilling in me the desire for higher education and the confidence in myself to know that I could be successful in any field I chose. To my father for always supporting me and showing active interest in my studies. To my mother for providing an environment in which study was possible.

Sabine Bergler, my thesis advisor and friend, was the one who first suggested to me that I was capable of completing a master's degree in computer science. She has been very understanding of the many challenges and obstacles that I faced during the course of writing my thesis. Sabine, I thank you for your support and I hope that our friendship will continue to grow.

Thank you to my colleagues at Motorola. Brian's interest in my thesis renewed my drive to get it finished. His comments and encouragement arrived at a very critical time. Thank you to Russ and Geoff for giving me the time I needed to get back into writing my thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Information retrieval either on the Internet or in document databases can often lead to hundreds of documents being selected of which only a small number are of any interest to the user. Such searches today are usually based on simple pattern matching of keywords or phrases. Often, the user can specify that the match should occur in the title, the author's name, or the text. Homonyms, annotations and small references to the pattern are all reasons why many searches are unsuccessful or where the important documents are buried in mounds of unrelated documents.

Some effort has been made in the field of information retrieval to develop more "intelligent" ways of searching databases and the Internet such as web-bots and agents (see [Bradshaw, 1997, Chin, 1991] for introductions to software agents and intelligent interfaces), with varying degrees of success. For any search method to be truly effective, the tool doing the search needs to have some knowledge of the user. In order to identify the subset of documents found that will be of interest to the user, the search tool needs to do much more than simple pattern matching. Moukas [Moukas, 1997] proposes Amalthea, which he describes as a multi-agent evolving ecosystem, to discover and filter documents on the Internet. It serves to tailor search results to the user's interests which are obtained through a direct feedback mechanism in which the user indicates the relevance of each retrieved document. However, its filtering strategy is still essentially based on keywords, albeit a more complex type of keyword called an augmented keyword vector. This is sufficient for basic information gathering. But, for tasks where diverse knowledge is to be interpreted or summarised, it is useful to have some way of representing what the user believes, some semantic understanding of the documents it is searching through, and some way of knowing what the user is looking for and why. For example, the user might be searching for documents which support a particular belief of the user, or conversely, documents which contradict or oppose the user's beliefs on a topic. It is also possible that a user has no preconceived beliefs about a topic and is looking to gather varied information on a topic. For example, a foreigner who has no particular stake on the issue of Quebec separation might want to gather information from both the "Yes" and "No" camps.

Supplied with the model of a user's beliefs, a search tool would also be better able to determine what are the appropriate types of documents to be retrieved. For instance, it would be appropriate for a search tool to retrieve an introduction to cellular systems for a novice trying to gather information but to retrieve cellular specifications from standards bodies for a domain expert.

This thesis addresses the issue of how a model of a user's beliefs can be created by simulating the belief acquisition of readers of news articles. We assume that when a person reads a news article, that person acquires information from the article. By simulating how a person wants to acquire information from an article, a model representing the beliefs of the person can be created. Based on the acquired information, the model can be used to predict which articles will be of more interest to the user. With each article that the user chooses to read, the simulation process is repeated. The model grows and represents more and more of the user's beliefs allowing the system to make more precise predictions about which documents should be retrieved during a search.

1.2 What is Belief?

In order to build a model of a user's beliefs, it is important to define what *belief* means in this context. Does *belief* differ from *fact*? Traditionally, in the field of Artificial Intelligence, a fact has often been defined as *justifiable true belief*. By consequence, a belief is something that is not necessarily justifiable or true. However, this is not a sufficient definition of *belief*. The Collins Concise Dictionary Plus (CCDP) defines *belief* as:

> **belief**... 1. a principle, etc., accepted as true, esp. without proof. 2. opinion; conviction. 3. religious faith. 4. trust or confidence, as in a person's abilities, etc.

The Webster's Revised Unabridged Dictionary, 1913 defines *belief* as:

> **belief**... 1. assent to a proposition or affirmation, or the acceptance of a fact, opinion, or assertion as real or true, without immediate personal knowledge; reliance upon word or testimony; partial or full assurance without positive knowledge or absolute certainty; persuasion; conviction; confidence; as, belief of a witness; belief of our senses. 2. (Theol.) a persuasion of the truths of religion; faith. 3. the thing believed; the object of belief. 4. a tenet, or the body of tenets, held by the advocates of any class of views; doctrine; creed.

The common thread in the two definitions is that a belief is something that is *accepted as true* and that there is a lack of evidence or *personal knowledge*. The Webster's definition includes the notion that it is something that a person is confident about but does not know positively or with absolute certainty.

The CCDP defines *fact* as:

> **fact**... 1. an event or thing known to have happened or existed. 2. a truth verifiable from experience or observation.

Webster's 1913 also defines *fact* as:

> **fact**...1. a doing, making, or preparing. 2. an effect produced or achieved; anything done or that comes to pass; an act; an event; a circumstance. 3. reality; actuality; truth. 4. the assertion or statement of a thing done or existing, sometimes, even when false, improperly put, by a transfer of meaning, for the thing done, or supposed to be done; a thing supposed or asserted to be done; as, history abounds of false facts.

The CCDP definition for *fact* is very much in line with the traditional Artificial Intelligence definition. On the other hand, the Webster's definition, gives voice in sense 4. to the reality that so-called facts, are sometimes proven to be wrong.

What then does it mean to believe something as opposed to knowing something? Is there a difference? Is the difference important in the context of a model of a user's beliefs? The CCDP defines *to believe* in its transitive sense as:

> **believe**...1. to accept (a statement or opinion) as true or real: I believe God exists. 2. to accept the statement or opinion of a person as true. 3. to think, assume, or suppose. 4. to think that someone is able to do (a particular action): I wouldn't have believed it of him.

Webster's 1913 defines *to believe* in its transitive sense as:

> **believe**...1. to exercise belief in, to credit upon the authority or testimony of another, to be persuaded of the truth of, upon evidence furnished by reasons, arguments, and deductions of the mind, or by circumstances other than personal knowledge, to regard as, accept as true; to place confidence in; to think; to consider as, to believe a person, a statement, or a doctrine.

The CCDP defines *to know* as:

> **know**...1. to be or feel certain of the truth or accuracy of (a fact, etc.). 2. to be acquainted or familiar with: she's known him five years. 3. to have a familiarity or grasp of: he knows French. 4. to understand, be aware of, or perceive (facts, etc.): he knows the answer now. 5. to be sure or aware of (how to be or do something). 6. to experience, esp. deeply: to know poverty. 7. to be intelligent, informed, or sensible enough (to do something). 8. to be able to distinguish.

Webster's 1913 also defines *to know* as:

> **know**...1. to perceive or apprehend clearly and certainly; to understand; to have full information of; as, to know one's duty. 2. to be convinced of the truth of; to be fully assured of; as, to know things from information.

The common thread in the definitions of *to know* is that there is a ***certainty as to the truth*** of the thing known. On the other hand, *to believe* indicates an ***acceptance of the truth*** of something based on evidence other than personal knowledge.

Both definitions of *to believe* match very tightly with what is proposed in this thesis. A reader of news articles who acquires information from that article will *believe* (or not) what has been read as opposed to *knowing* what has been read. The new beliefs are accepted as true based on evidence other than personal knowledge. In this case, the evidence comes from the sources and the reporter. What is believed may be a *fact* but the reader can not call it so since he has no direct personal evidence of its truth or reality.

Interestingly, the original version of the Webster's Dictionary, published in 1828, went even further in its definition of *to believe* adding:

> When we believe upon the authority of reasoning, arguments, or a concurrence of facts and circumstances, we rest our conclusions upon their strength or probability, their agreement with our own experiences, etc.

It also added a caveat in its definition of *to know* warning that:

> To know a thing precludes all doubts or uncertainty of its existence. We know what we see with our eyes or perceive by other senses. We do not know the truth of reports, nor can we always know what to believe.

In the context of user modelling the distinction between fact and belief is an important one. As previously mentioned, the reader of news articles is acquiring beliefs about what is read. Even more explicitly, the reader is acquiring beliefs about what he believes the reporter believes the sources believe about something. These beliefs may be wrong. The reader may have misunderstood what was written. The reporter may have misquoted or misinterpreted the source. The source may be lying or have misinterpreted the event or situation being reported. Any system dealing with beliefs must also be able to deal with a change in the truth value of the belief. In theory, a system dealing with facts will never need to deal with a change in truth value since a fact is something that has been observed and is known to be true, *justifiable true belief*.

1.3 The Problem

Internet search engines are based on simple pattern matching mechanisms that yield hundreds of hits in a typical search. Important documents on the requested topic are often missed or are buried among the hundreds of useless or off-topic documents. Many of the documents found may be inappropriate for the user. They may be too superficial or too complex. One approach to limiting the searches to documents which are appropriate to the current user is to create a model of the user. The model must be dynamic. That is to say, it must be able to acquire information, to *learn*.

To be successful, the model should be created with as little intervention on the part of the user as possible. Asking the user questions about his likes and dislikes will prove to be quite tedious for the user, and will probably not help limit the search very much. The manner in which the model can be increased depends on the application.

Whether the user is searching the Internet as a whole or newspaper web sites, most of the documents retrieved will be in the form of news articles or academic papers. News articles are of particular interest since there is such an abundance of them. One method in which the model can acquire information is to extract information from the articles themselves. The information in the news article is being reported by a reporter who obtained the information from some source or sources who may, in turn, have obtained the information from another source or sources. In this sense, the information is hearsay. The model should then contain information extracted from the articles that the reader could accept as true. Since the reader has no personal knowledge of the truth of this information, any information retained by the reader can be called a belief. Thus, what is needed is a model of the reader's beliefs. A system that builds this type of belief base would need to simulate how the reader acquires beliefs from news articles.

The problem that this thesis will address is *how can a system simulate how a reader acquires beliefs from news articles to build a model of the reader's beliefs?* Several issues must be considered when trying to solve this problem:

1. **Changing Nature of Beliefs**

   Since a belief is something that is accepted as true, it's truth value is not absolute. The reader could get new information that changes the truth value of one of his beliefs. What mechanisms are needed to deal with this type of situation?

2. **Text Representation**

   Is there a method of representing text that could simplify the extraction of beliefs from the text?

3. **Reliability**

   Since the information in news articles consists of *reported speech*, it's reliability can be questionable. Are there factors that could be useful in determining the reliability of a belief?

4. **Representation of the Model**

   The model of a reader could potentially contain thousands upon thousands of beliefs. In order for the model to be of any use to a system, the beliefs must be represented and classified in a manner that will facilitate searching.

5. **Acquisition**

   How does a reader acquire beliefs from reading? What are the mechanisms that a system could implement to approximate the acquisition? All readers are different and some might have previous beliefs about certain topics. These previous beliefs need to be taken into consideration when acquiring beliefs.

This thesis outlines one approach to this problem that takes into consideration all of these issues.

1.4 The Solution

Given all the issues outlined in the previous section, the solution to this problem incorporates many components. This section will briefly discuss the various components and how together they solve the problem *how can a system simulate how a reader acquires beliefs from news articles to build a model of the reader's beliefs?*

As previously discussed, beliefs are propositions that have been accepted as true. In the case of the belief model of a reader of news articles, they are propositions from ***reported speech*** which the reader has accepted as true. The reader has no personal knowledge or evidence of the veracity of the propositions so they may, in reality, not be true. It is possible that the reader may read another article in which the proposition is found to be false. If the reader had already accepted the belief as true, he will need to change his belief. This may have an impact on other beliefs since the belief may have been a pre-requisite to another belief. Chapter 2 discusses systems known as ***Belief Maintenance Systems*** or ***Truth Maintenance Systems***. These are systems that can reason based on the truth value of propositions. And, more importantly, they can determine if a contradiction exists in the system. A contradiction can occur when the truth value of a proposition changes. The job of such a system in this situation, is to identify all the propositions that are affected by or contributed to the contradiction and re-evaluate them. In some cases, the belief will need to be removed. Any system that models the beliefs of readers of news articles will need to incorporate a belief maintenance component. Chapter 2 also introduces a framework for a new generation of belief maintenance system called ViewFinder [Ballim, 1992, Ballim and Wilks, 1992] that builds and maintains beliefs within environments called ***viewpoints***. Viewpoints can be created for every agent in a news article. ViewFinder and its partial implementation ***ViewGen*** are central to this thesis.

In order to facilitate the extraction of beliefs, news articles need to be represented in a manner that will put into evidence certain aspects of the text. One method, called profiles [Bergler, 1995a], will be used in this thesis. A ***profile*** is a grouping of all the propositions or statements made by a source in a news article. Many news articles contain information from sources with differing opinions. The profiles of sources who agree with each other are grouped into supporting groups and if there are two such supporting groups with differing opinions, they are called opposing supporting groups. An entire news article can be represented by a complex profile structure as discussed in Section 3.5. The complex profile structure can serve as a starting point for identifying the agents whose viewpoints need to be represented as well as the relationships between them.

In profiles [Bergler, 1992], important elements of the text such as the reporting verb and the lexicalisation of the source are separated from the reported information. A reporting verb is a verb that is used to report the speech of others. Verbs such as ***to say, to announce, to claim*** and ***to deny*** are examples of reporting verbs. Reporting verbs are very important for the acquisition of beliefs by readers of news articles because they contain information encoded by the reporter. This information indicates the reporter's level of confidence in the reported speech. Similarly, the lexical realisation of the source is also used by the reporter to encode his level of confidence in the source. The lexical realisation of the source is the manner in which the reporter refers to the source. The reporter can use the lexicalisation to indicate to the reader that the source should be believed because of his expertise in the topic, his involvement with the topic or his authority to speak about the topic. Chapter 3 introduces a methodology for analysing reported speech called ***Evidential Analysis*** [Bergler, 1992]. By conducting an evidential analysis, the system is able to extract from the reporting verb and the lexicalisation of the source the information that was encoded by the reporter and determine the reporter's level of confidence in the source and in the reported information.

The *reporter's confidence in the reported information* as encoded by the reporting verb and the *reporter's confidence in the source* as encoded by the source description are two criteria that can be used by the reader (or a system modelling a reader) to determine the reliability of the belief. Two other criteria that can be used to determine the reliability of the belief are the *reader's confidence in the source* and the *reader's confidence in the reporter*. The evaluation of these last two criteria must be based on beliefs that the reader has about the source and the reporter if any.

The beliefs in the model need to be organised in such a way that the reader's beliefs about a certain topic or agent can be retrieved quickly and efficiently. If all the beliefs are lumped together in one big belief-base, the retrieval time will grow exponentially as the belief-base grows. One method, proposed by Ballim and based on work by Wilks and Bien is to compartmentalise beliefs according to the topic of the belief and according to the owner of the belief. For example, all of Mary's beliefs would be grouped together. Mary's beliefs would then be subdivided according to topic. All of Mary's beliefs about dogs would form one subgroup and all of her beliefs about cats would form another subgroup. Mary might also have beliefs about John's beliefs about dogs. These would form another subgroup of Mary's beliefs.

In the case of a system modelling the beliefs of a reader of news articles, the system's beliefs about the reader would be grouped together. The system's beliefs about the reader's beliefs would be a subgroup nested within the system's beliefs. This method of representing the model uses a diagrammatic tool called *belief diagrams*. Belief diagrams are a simple yet very effective method of representing viewpoints. Chapter 4 discusses how viewpoints are used in ViewFinder to represent the belief models of agents in a system.

In addition to providing a technique for representing belief models, ViewFinder also defines a mechanism, *belief ascription* [Ballim, 1992], by which beliefs can be acquired without direct evidence. It is based on the simple principle that, unless we have evidence to the contrary, we can assume that others have the same beliefs as us. Chapter 5 discusses belief ascription and how it can be used to build models of agents and models of agents' models of other agents. It is important for an agent to be able to model another agent's beliefs in order to be able to communicate effectively and to understand the other's motives. A side-effect of belief ascription is belief percolation [Ballim, 1992]. It occurs when an agent has a belief that the system doesn't. If the agent is deemed to have some expertise about the subject of the belief, the system will acquire the belief from the agent.

In the case of a system simulating how a reader acquires beliefs from reading news articles, belief ascription is not necessary although it could be useful to simulate how a reader *interprets* what is read, adding and compounding his beliefs to those of the sources. Rather, a more interesting approach is to use the notion of belief percolation. Belief percolation in ViewFinder is discussed in Section 5.4.

Chapter 6 discusses how this thesis proposes to modify Ballim's *belief diagrams* to better protect the model from the even more questionable nature of beliefs obtained from news articles by introducing the concept of *potential belief* and *held belief*. Ballim's *belief percolation* is modified to be a stand-alone mechanism by which the reader acquires *potential beliefs* from other agents represented in the article. A method of establishing the reliability of a belief called a *belief heuristic* is introduced. The belief heuristic will determine the reliability of a potential belief based on four confidence criteria: the reporter's confidence in the source, the reporter's confidence in the reported information, the reader's confidence in the reporter, and the reader's confidence in the source as well as on the reader's previous beliefs. A tool called a ***source list*** is introduced. A source list, in the context of this system, contains information about the sources of the belief, the reporter who reported the belief, and the evaluations of the four confidence criteria. The source list is also used by the belief heuristic. A new mechanism, called ***belief promotion*** is used to promote potential beliefs to held beliefs. Potential beliefs are promoted when they have been deemed reliable by the belief heuristic.

Chapter 7 discusses a system called *Percolator*. *Percolator* implements all the mechanisms and modifications described in Chapter 6. It is used to build Ballim-type environments as described in Section 4.2. These environments represent the viewpoints of the agents in news articles about the topic of the news article. The most important mechanisms presented in this thesis and implemented in *Percolator* are the percolation mechanism, the decomposition mechanism, and the promotion mechanism guided by a belief heuristic. The chapter on *Percolator* describes the design of the system, the classes used in the system, as well as the algorithm of the implemented mechanisms.

Chapter 8 shows how belief percolation as implemented in *Percolator* can be used by a system to simulate how a reader acquires beliefs from news articles in order to build a model of the reader's beliefs. Three types of reader models are defined: the naive reader with no previous beliefs about the topic, the reader with supporting beliefs about the topic, and the reader with opposing beliefs about the topic. A sample text, taken from the web site of a newspaper, is used as an example to show how all the components are needed to percolate beliefs from the news article into the reader's viewpoint.

Chapter 2

Belief Revision

Systems that need to communicate with other agents need to have some understanding of the other agents' motives or plans. In such systems, agents can be computers, other systems or humans. In order to communicate with the other agent, the system needs to have a model of the agent. This model should contain information about the agent. According to the discussion in Section 1.2, this information consists of *beliefs* since much of it is based on assumptions about the other agent.

One of the most basic problems in agent modelling is that of handling a change in the beliefs in the model of an agent. Changes in belief will often lead to a contradiction in the model making it inconsistent. Researchers in the field have developed a number of strategies for dealing with belief change [Friedman and Halpern, 1994, Boutilier and Becher, 1995, Liberatore and Schaerf, 1996, Li and Pereira, 1996]. However, *belief revision* [Doyle, 1991, Alchourrón *et al.*, 1985] remains the most used strategy. Belief revision in agent modelling consists of three main tasks: first, detecting situations in which new information is inconsistent with the beliefs in the model; second, deciding whether to reject the new information or somehow modify the agent's beliefs to match with the new information; third, in the case that the new information is accepted, the belief that caused the inconsistency must be found and removed from the model.

More advanced systems will be able to perform *rational belief revision* by which the system will be able to choose between several possible alternative revisions based on notions of preference and epistemic entrenchment [Doyle, 1991, Gärdenfors, 1988, Nebel, 1989] such that belief revisions will attempt to retain those beliefs that have the most dependencies. A system that performs belief revision on a database of information, whether it be a user model (belief base), a knowledge base or a database, is called a *reason maintenance system*.

2.1 Overview of Reason Maintenance Systems

A reason maintenance system (RMS) is, essentially, a system that supports a general reasoning system. The reasoning system will provide inferences to the RMS, usually in the form of propositional arguments for statements called *justifications*. It is the purpose of the RMS to maintain a *dependency theory* of the justifications that connect the propositions called *beliefs*. Witteveen and Brewka [Witteveen and Brewka, 1993] view the tasks of an RMS as consisting of *basic reason maintenance* and *belief revision*. Basic reason maintenance entails ensuring "a coherent and consistent interpretation of the beliefs in the dependency theory and to update the belief status after the addition of new justifications." [Witteveen and Brewka, 1993, p.2] However, as Doyle [Doyle, 1983] contends, consistency is maintained by the RMS only in the sense that the assumptions made by the RMS, the system's beliefs, must be consistent with the justifications held by the system. Belief revision, then, consists of performing conflict resolution when a contradiction or inconsistency has been detected in the set of current beliefs.

Research in the domain of RMSs has gone from systems based on binary truth-values to systems based on probabilistic truth-values. Truth maintenance systems (TMSs) (recently they are being referred to as *Reason Maintenance Systems* [Doyle, 1983]) are stand-alone reasoning components that incrementally maintain the beliefs in a system allowing it to reason with partially specified knowledge or assumptions. Three types of TMSs have been developed. The first type is the *Justification-based* TMS (JTMS). A JTMS records dependencies between the propositions and allows reasoning based on the assumption of some of the propositions. When a contradiction is detected, it backtracks through the chain of dependencies and identifies the assumptions that caused the contradiction. Doyle's *Truth Maintenance System* [Doyle, 1979] is an example of a JTMS. The second type of TMS is the *Assumption-based* TMS (ATMS) introduced by de Kleer [de Kleer, 1986]. ATMSs label propositions with the minimal consistent set of assumptions that can be used to prove them. The third class of TMS is the *Logic-based* TMS (LTMS). LTMS do not work with dependencies among propositions. Rather, propositions and truth-functional connectives are used to build full propositional formulas. Truth values are then propagated using the *Boolean Constraint Propagation* algorithm [McAllester, 1990].

Recently, efforts have been made to introduce probability into the different types of TMS. A TMS that incorporates aspects of probability is known as a *Belief Maintenance System*. Falkenhainer [Falkenhainer, 1986] introduced probabilities into a JTMS creating a Justification-based BMS (JBMS). The same has been done with the ATMS. Ramoni and Riva [Ramoni and Riva, 1993] have added probabilistic logic to the LTMS to develop a LBMS in which constraints on the probabilistic truth-values of propositions are provided by the boolean operators of standard logic.

Some in the field are beginning to view the task of belief revision in terms of probabilities, so the problem then becomes one of the revision of probability assessments. These systems are called *Bayesian Belief Networks* (BBNs). One of the shortcomings of such systems is that due to the nature of Bayesian networks, the systems are incapable of providing adequate explanations of their reasoning. More recently, attempts have been made to give the BBNs the capability of providing explanations as to the decisions they make by combining the properties of the LBMS and the BBN to form what is called an *Ignorant Belief Network* (IBN) [Ramoni and Riva, 1994].

In domains that require a system to participate in a dialogue or to perform cooperative planning with other agents, it is often required to build and maintain complex models of the beliefs of agents with whom it is interacting. Very little research has been done on the problem of belief revision in multi-agent systems [Florea, 1997], particularly in systems that maintain nested models of agents (an agent's model of another agent). One notable effort in this field is that of Ballim and Wilks [Ballim, 1992, Wilks and Ballim, 1987, Ballim and Wilks, 1991, Ballim and Wilks, 1992].

Section 2.2 gives a general discussion of Truth Maintenance Systems. It provides a brief explanation of the three types of truth maintenance systems and takes a closer look at Doyle's TMS which implements a JTMS. Section 2.3 will look at Belief Maintenance Systems and, more particularly, a BMS framework called ViewFinder [Ballim, 1992]. ViewFinder is a framework for representing, ascribing and maintaining the nested beliefs of interacting agents. It introduces an entirely new type of representation and uses a form of default reasoning to generate belief spaces based on information encountered without assumptions of completeness. ViewFinder will serve as the basis for the solutions presented in this thesis.

2.2 Truth Maintenance Systems

Research in the domain of truth maintenance has centred around justification-based truth maintenance systems, assumption-based truth maintenance systems and to a lesser degree logic-based truth maintenance systems. Martins and Shapiro [Martins and Shapiro, 1986] assert that a fundamental issue for all systems performing belief revision is that they must be able to determine each and every proposition which could have contributed to a contradiction. This is done by keeping a record that lists all the other propositions that a proposition is based on. This list or set is called the support of the proposition. In this way, when a contradiction is discovered, the system has a way of identifying all the possibly affected beliefs.

One important way in which the three types of truth maintenance systems differ is in the contents of the support set. In justification-based systems, the support set contains the list of propositions that directly led to the proposition. In assumption-based systems this list contains the hypotheses or non-derived propositions that led to the proposition [Martins and Shapiro, 1986].

A second important way in which they differ is in their search orientation. The JTMSs and LTMSs use a depth-first mechanism which will find one solution to a given problem. The ATMSs, on the other hand are oriented towards finding all possible solutions [de Kleer, 1986] using a breadth-first search mechanism.

A third important way in which they differ is in their approach to belief revision. Two belief revision approaches have been greatly studied in recent years: the *foundations approach* and the *coherence approach*. (See [Doyle, 1992] for a good comparison of the two approaches.) According to the foundations approach, a rational agent will derive its beliefs from justifications or reasons. That is to say, an agent will only hold a belief if it also has a good reason for that belief. If a reason or justification is abandoned or adopted, the agents beliefs may change. The coherence approach, on the other hand, argues that the reasons or justifications for a belief are of little importance. An agent will hold a belief as long as it is coherent and consistent with its other beliefs. Furthermore, belief revisions should change only the minimal set of beliefs necessary to maintain consistency as specific beliefs are added or removed from the system. Doyle's *Truth Maintenance System* [Doyle, 1979] is seen as exemplifying the foundations approach on a JTMS backbone. Whereas, the AGM theory [Alchourrón *et al.*, 1985] is seen as exemplifying the coherence approach on an ATMS backbone.

Most truth maintenance systems employ as part of their foundation a formal model of knowledge and belief proposed by Hintikka [Hintikka, 1962] called the ***possible worlds*** approach. The possible worlds approach states that given W, a set of alternative worlds, the knowledge and belief of an agent situated in the current world w "consists of that agent identifying a subset $W\prime$ of W as the set of worlds that are possible alternatives to w" [Vardi, 1986]. That is, a proposition p is known or believed in w by a if p is true in all the worlds in $W\prime$.

As claimed by many in the field [Vardi, 1986, Konolige, 1986, Hintikka, 1975], there is a serious flaw in this theory which has been given the name ***logical omniscience***. Logical omniscience means that an agent must always know or believe the logical consequences of her knowledge or beliefs. In other words, if p logically implies q and the agent knows or believes p then the agent also knows or believes q. This does not model belief in humans very accurately.

Fagin and Halpern [Fagin and Halpern, 1985] and Levesque [Levesque, 1984] attempt to deal with this issue by distinguishing between ***implicit*** and ***explicit*** beliefs. Explicit beliefs are beliefs that the agent has or claims to have whereas implicit beliefs are all the logical consequences of the explicit beliefs [Konolige, 1986]. Fagin and Halpern call this the ***logic of general awareness***. It means that if p logically implies q and the agent knows or believes p then the agent also knows or believes q but is ***unaware*** of this. This does not adequately represent human belief systems either.

2.2.1 Justification-based Truth Maintenance Systems

One of the first justification-based truth maintenance systems (JTMSs) is Doyle's ***Truth Maintenance System*** (TMS) [Doyle, 1979]. Doyle made the observation that, as humans, we continually make assumptions about objects, their permanence, features, and properties and that we can make corrections to our assumptions with relative ease. We can easily provide explanations for our errors and replace our old conclusions with new ones based on new evidence. This observation shows that for humans, the set of our beliefs changes in a ***non-monotonic*** manner.

The purpose of the TMS was to show how non-monotonic reasoning could be used to "determine the current set of beliefs from the current set of reasons, and to update the current set of beliefs in accord with new reasons in a (usually) incremental fashion." [Doyle, 1979, p.231]

Reasoned Belief

In TMS, the reason (or justification) for a belief consists of an ordered pair of sets of other beliefs. A reasoned belief is ***in*** only if every belief in the first set is ***in*** and every belief in the second set is ***out***. An ***in*** belief is one that is believed. Consequently, an ***out*** belief is one that is not believed. To be ***in*** a belief must have at least one currently valid reason, and to be ***out***, the belief must have no currently valid reasons or no reasons at all. An assumption is a current belief (an ***in*** belief) whose reason depends on a non-current belief (an ***out*** belief).

Doyle considers three main types of justification. The first is ***non-monotonic justification*** used to make tentative guesses. Here, an argument for a belief (represented as a node in TMS) can be based on current belief in other nodes or on lack of current belief in other nodes. For example, given the node N-1 representing the statement P and the node N-2 representing $\neg P$, N-1 can be justified on the basis of lack of belief in N-2. The TMS can keep N-1 as a current belief as long as N-2 is not a current belief. The TMS has assumed belief in N-1. An assumption can then be described as a node whose well-founded support is a non-monotonic justification.

The second type of justification is the *support-list (SL) justification*. It has the form:

$$(SL\langle inlist\rangle\langle outlist\rangle)$$

The SL-justification is a valid reason for a belief if and only if each of the nodes in the first list is believed and each of the nodes in the second list is not believed.

The third type of justification is *conditional-proof (CP) justification*. A conditional-proof justification has the form:

$$(CP\langle consequent\rangle\langle INhypotheses\rangle\langle OUThypotheses\rangle)$$

The justification is valid if the consequent node is *in* when each node in the INhypotheses list is *in* and each node in the OUThypotheses list is *out*. Whenever the TMS finds that a particular CP-justification is valid, it computes an equivalent SL-justification. SL-justifications (derived or otherwise) are the only justifications that constitute supporting justifications for nodes.

Belief Update

To update beliefs the TMS uses *reasoned retraction of assumptions*. This method states that an assumption is retracted by giving a reason why it should be retracted. If at a later time the reason for retraction proves to be invalid, the assumption is restored. This means that to retract an assumption you have to make it *out*.

From the reasoned retraction of assumptions, procedures were developed for revising the current set of beliefs when inconsistencies are found. One such procedure used by TMS is *dependency-directed backtracking*. When the TMS finds an inconsistency in beliefs represented by the certain current beliefs, it uses these beliefs as arguments for a new belief and marks it as a contradiction. When the TMS finds a contradiction node that is *in*, it invokes the dependency-directed backtracking procedure. The procedure will find and discard at least one current assumption which will in turn make the contradiction node *out*.

The Truth Maintenance Process

The truth maintenance process in TMS is responsible for making all necessary changes in the current set of beliefs whenever a belief is added or removed. The algorithm is summarised in the following 7 steps:

1. Add a new justification

   The process is invoked when a new justification is added to a node.

2. Update the beliefs required

   Some bookkeeping is needed if the new justification is invalid or if the node is already *in*.

3. Mark the nodes

   If the new justification is valid and the node is *out*, the system must update the node and its repercussions. The system makes a list of the node and its repercussions and marks each of these nodes to indicate that they have not been given well-founded support. (A node is said to have well-founded support if one of its justifications consists of non-circular arguments (premises)).

4. Evaluate the nodes' justifications

   The TMS evaluates the justifications to see if any have well-founded support. If any are found, they are made to be *in* or *out* as the case may be. The marked consequences are evaluated to see if any of them have well-founded support.

5. Relax circularities

   Sometimes, not all the nodes receive well-founded support in the previous step. In such cases, a constraint-relaxation process is invoked. This process assigns support-status to these nodes.

6. Check for CP-justifications and contradictions

   In this step, the TMS checks for contradictions and invokes the dependency-directed backtracking and CP-justification procedures when necessary.

7. Signal changes

   Informs the user of the changes in support-status of the nodes.

Doyle uses an example to show how the truth maintenance process works. A program has been asked to schedule a meeting with the preference that it be held at 10:00 am in room 813 or 801. This data is represented in Table 1 as N-1, N-2, N-3, and N-4. Given that the system only has these four beliefs, it makes N-1 and N-3 *in* and N-2 and N-4 *out*.

What happens if the system discovered that a meeting had already been scheduled for 10:00 am in room 813? It creates a new node, N-5, and uses N-1 and N-3 as part of its justification. The new node is then declared a contradiction. Because there is a contradiction in the belief set, the dependency-directed backtracking procedure is invoked. It traces backwards through the justifications for the contradiction, N-1 and N-3. N-1 is justified by N-2 and N-3 is justified by N-4. Since N-1 and N-3 are justified independently they are said to be maximal[1]. N-6 is created using conditional proof justification with N-1 and N-3 in the INhypotheses list. N-3 is arbitrarily chosen to be the cause of the contradiction and so it is made to be out by making one member of it's out list to be *in*. N-3

[1]The dependency-directed backtracking will find the *maximal* assumptions by tracing through the foundations of the contradiction node C to find the set $S = \{A_1, \ldots, A_n\}$, which contains an assumption A if and only if A is in C's foundations and there is no other assumption B in the foundations of C such that A is in the foundations of B. S is then the set of maximal assumptions [Doyle, 1979]

N-1	TIME(M) = 1000	(SL ()(N-2))	
N-2	TIME(M) ≠ 1000		
N-3	ROOM(M) = 813	(SL ()(N-4))	
N-4	ROOM(M) = 801		
N-5	CONTRADICTION	(SL (N-1 N-3)())	
N-6	NOGOOD N-1 N-3	(CP N-5 (N-1 N-3)())	*here* ≡ (SL ()())
N-4	ROOM(M) = 801	(SL (N-6 N-1)())	
N-7	CONTRADICTION	(SL (N-4)())	
N-8	NOGOOD N-1	(CP N-7 (N-1)())	*here* ≡ (SL (N-6)())
N-2	TIME(M) ≠ 1000	(SL (N-8)())	

Table 1: An Example of TMS: Scheduling a meeting

only has one node in its out list, N-4. N-4 is made to be *in* by virtue of nodes N-6 and N-1. At this point, N-1, N-4, and N-6 are *in* and N-2, N-3, and N-5 are *out*.

Suppose now that the system discovers that room 801 can not be used. The system creates a new contradiction N-7 which is supported by N-4. It creates N-8 with a CP-justification using N-1 in the INhypotheses list. This is equivalent to using the support list justification with N-6 in the *in* list because N-7's foundations contain N-6 and N-1's repercussions do not. At this point N-1 is no longer believed, so N-4 is also *out*. The *in* nodes are now N-2, N-3, N-6, and N-8 which means that the meeting will be held in room 813 at a time other than 10:00 am.

2.2.2 Assumption-based Truth Maintenance Systems

De Kleer's assumption-based truth maintenance system (ATMS) was a very important addition to the field of truth maintenance for many reasons. De Kleer [de Kleer, 1986] claims that it overcomes many of the disadvantages or limitations of the justification-based truth maintenance systems.

One important feature of the ATMS is the introduction of labels. Each datum is labelled with sets of assumptions representing the contexts under which it holds. Assumptions are the primitives from which all other data are derived. The ATMS then manipulates the assumption sets as opposed to the datum sets they represent. There is no need for the database to be consistent. Referring to other contexts and moving to other points in the search space are relatively simple tasks.

In an ATMS, each node contains the following information:

- the datum
- the label
- the justifications

Each node has the following form:

$$\gamma_{datum} : < datum, label, justifications >$$

There are four types of nodes: premises, assumptions, assumed nodes, and derived nodes. A premise is a node that does not need to be justified. It holds universally. The node

$$< p, \{\{\}\}, \{()\} >$$

represents the premise p. An assumption is a node that is self-justified. The label contains a singleton mentioning itself. An assumption can also contain justifications. The node

$$< A, \{\{A\}\}, \{(A), (d)\} >$$

represents the assumption A. In this case, the assumption A is justified by itself or the node d. It holds in the environment $\{A\}$. An assumed node is not a premise nor an assumption. It has a justification mentioning an assumption. The node

$$< a, \{\{A\}\}, \{(A)\} >$$

represents the assumed node a. It is justified by the assumption A and holds in the context of A. All other nodes are derived. The node

$$< w = 1, \{\{A, B\}, \{C\}, \{E\}\}, \{(b), (c, d)\} >$$

states that $w = 1$ can be derived from the node b or the nodes c and d. It also states that the node holds in environments $\{A, B\}$ and $\{C\}$ and $\{E\}$.

Also, like the JTMS, the ATMS has *nogood* environments. These are inconsistent environments that contain inconsistent conjunctions of assumptions. For example, *nogood*$\{a, b\}$ means that a and b can not be true at the same time.

Consider the following sequence of clauses:

assume A

assume B

assume C

assume D

assume E

nogood$\{A, B, E\}$

$A \wedge B \rightarrow p$

$B \wedge C \wedge D \rightarrow p$

$A \wedge C \rightarrow q$

$D \wedge E \rightarrow q$

An ATMS would create the following labels:

$$< A, \{\{A\}\}, \{(A)\} >$$

$$< B, \{\{B\}\}, \{(B)\} >$$

$$< C, \{\{C\}\}, \{(C)\} >$$

$$< D, \{\{D\}\}, \{(D)\} >$$

$$< E, \{\{E\}\}, \{(E)\} >$$

$$nogood\{A, B, E\}$$

$$< p, \{\{A, B\}, \{B, C, D\}\}, \{(A \wedge B \rightarrow p), (B \wedge C \wedge D \rightarrow p)\} >$$

$$< q, \{\{A, C\}, \{D, E\}\}, \{(A \wedge C \rightarrow q), (D \wedge E \rightarrow q)\} >$$

At this point, the ATMS is given the clause $p \wedge q \rightarrow r$. The ATMS will need to update its beliefs. That is to say, it will need to perform belief revision as follows:

- Find the labels of the nodes in the clause. This will give:

$$p : \{\{A, B\}, \{B, C, D\}\}$$

$$q : \{\{A, C\}, \{D, E\}\}$$

- Find all the environments in which the new node can hold. Since there are two assumptions (p and q) there are $2^n = 4$ environments.

$$\{\{A, B, C\}, \{A, B, D, E\}, \{A, B, C, D\}, \{B, C, D, E\}\}$$

- Eliminate any *nogood* environments. The environment $\{A, B, D, E\}$ is eliminated since it is a superset of the *nogood* environment giving

$$\{\{A, B, C\}, \{A, B, C, D\}, \{B, C, D, E\}\}$$

- Eliminate all redundant environments. The environment $\{A, B, C, D\}$ is eliminated since it is a superset of environment $\{A, B, C\}$ giving

$$\{\{A, B, C\}, \{B, C, D, E\}\}$$

- Associate the new label to the node r.

$$< r, \{\{A, B, C\}, \{B, C, D, E\}\}, \{(p, q)\} >$$

The limitations of the justification-based truth maintenance systems (JTMS) that have been overcome by assumption-based truth maintenance systems (ATMS) as seen by de Kleer [de Kleer, 1986] are the following:

1. Single state problem

   The JTMS only allows one solution to be considered at a time. In ATMS, there is no current global context so multiple, mutually contradictory solutions can exist at one time.

2. Overzealous contradiction avoidance

   The JTMS will guarantee that given two contradictory beliefs, only one will be taken into consideration. The ATMS, on the other hand, will guarantee that other beliefs that depend on *both* of the contradictory beliefs will not be considered.

3. Switching states is difficult

   In JTMS, there is no way to temporarily change an assumption. An assumption can be changed by introducing a contradiction but since the contradiction can never be removed, the knowledge state is permanently altered. In the ATMS, the problem of switching states is irrelevant since states are specified by a set of assumptions. The system can search in one context (set of assumptions, environments) or in all contexts simultaneously.

4. Dominance of justifications

   Assumptions as defined by Doyle [Doyle, 1979] are nodes whose supporting justifications depend on another node being *out*. During a problem-solving task, support justifications may change so the set of nodes considered to be assumptions changes as well. In ATMS, assumptions are manipulated not justifications. Therefore, the justifications supporting data never change.

5. Cumbersome Machinery

   The truth maintenance algorithm as described in 2.2.1 tries to find a solution that can satisfy all the justifications. This in addition to the determination of well-founded support and the dependency-directed backtracking process are very involved and time-consuming. The status of the data can also change many times between *in* and *out* during this process. The machinery behind the ATMS is considerably simpler since no backtracking is ever necessary and because the assumptions supporting a contradiction are directly identifiable.

6. Unouting

   During problem solving, data can be derived, made *out* when a contradiction occurs, and made *in* again (reasserted) when a second contradiction changes the context. Unouting is the process used to decide which data can be reasserted. The problem in JTMS is that sometimes data is re-derived. In ATMS, all solutions are explored simultaneously, so there is no need for backtracking, retraction, or context switching. The unouting problem does not exist in the ATMS.

Like JTMSs and LTMSs, ATMSs are based on the *possible worlds* approach. The assumption is one of *deductive closure* which means that everything derivable from the set of propositions in the system is also known by the system. Unfortunately, it is not feasible to have a deductively closed real-world application. The amount of storage space would be enormous and the search time would be beyond acceptable limits. To deal with this, Johnson and Shapiro [Johnson and Shapiro, 2000], inspired by integrity constraints first identified by Gärdenfors and Rott [Gärdenfors and Rott, 1995], implemented a modified set of integrity constraints in a deductively open belief space ATMS. The integrity constraints allow the system to reason even though the system cannot guarantee deductive closure.

2.2.3 Logic-based Truth Maintenance Systems

The logic-based truth maintenance system (LTMS) was first introduced by McAllester [McAllester, 1978, McAllester, 1990]. Unlike JTMSs and ATMSs, LTMSs do not focus on the dependencies between propositions in the set of beliefs. Rather, they manipulate proposition symbols and *boolean constraints*. A boolean constraint is any boolean formula containing proposition symbols and standard boolean connectives (i.e. $\rightarrow$ (implication), $\wedge$ (conjunction), and $\neg$ (negation)). Each constraint represents a logical constraint on the truth-values of the propositions it contains.

The main contributions made by McAllester [McAllester, 1978, McAllester, 1990] to the field of truth maintenance is that LTMSs allow for *three valued truth maintenance*. In JTMSs and ATMSs, the truth values of propositional symbols are binary. In an LTMS, each proposition symbol can have the value true, false, or unknown. Each proposition in the clause has a label reflecting whether it is negated or unnegated in the clause. At least one proposition in the clause must have a truth-value consistent with its label in the clause in order for the clause to be satisfied. Furthermore, once a constraint has been added to the system it can not be removed.

A literal, in LTMS terms, is a proposition symbol or the negation of a proposition symbol. The set of literals is called a premise set. A special proposition symbol, contradiction, is used to inform the user that a given premise set is inconsistent with the set of boolean constraints. A contradiction can occur if a proposition is labelled as *true* and *false* or if a clause (constraint) is completely violated meaning that all propositions in the clause have a truth-value opposite to their label in the clause.

One of the main mechanisms in an LTMS is known as *Boolean Constraint Propagation* [McAllester, 1990]. Consider a network in which the nodes are propositional symbols. The symbols are either in the premise set Σ or in the boolean constraint set. The boolean constraints in the set represent connections or relationships between the nodes in the network. Each node can have the label true, false, or unknown. All the nodes are initially labelled unknown.

The consequences of a premise set Σ can be calculated by assigning the label true or false to each proposition symbol in the premise set Σ. The label will be true if the proposition symbol is unnegated in Σ and false if it is negated in Σ.

New labels can be calculated based on *local propagation*. If a new truth value follows from existing labels *and* a *single* internal constraint, the new label is added to the network and propagation continues. A set of derived labels is said to *violate* an internal constraint when all proposition symbols in the constraint have a truth-value opposite to their label in the constraint. If a constraint is violated, then the special proposition contradiction is labelled true. A label derived during the propagation process can be associated with a justification – the labels and the constraint used to derive it.

According to McAllester [McAllester, 1990], any LTMS must be able to do the following:

- Add a boolean constraint
- Determine if a literal follows from a given premise set and the set of boolean constraints
- Provide a set of literals such that each literal is derivable from a given premise set and the set of boolean constraints
- Provide a subset of the constraints such that one or more literals is derivable from a given premise set and the subset of constraints and that each literal follows from a given premise set and the subset of constraints
- Determine the set of literals and the set of constraints that have participated in a contradiction

The first two items are part of the propagation process. The third and fourth items enable the LTMS to justify or "prove" a derived proposition symbol. The final item enables the LTMS to detect a contradiction within the set of proposition symbols in the system.

In real world applications, the size of the cache of beliefs (nodes) and inferences in truth maintenance systems grows very quickly with each addition or retraction of a belief. In fact, the cache can grow so quickly that the memory demand results in degraded performance and even system crashes. Many have designed more efficient search algorithms to try to decrease the search time. Everett and Forbus [Everett and Forbus, 1996], on the other hand, have developed an algorithm, called *fact garbage collection* (Fact-GC) that tries to identify facts that are unlikely to be needed again once retracted and that are almost as easy to derive using rules as they would be using truth maintenance operations. By deleting these facts, the size of the cache is reduced. Everett and Forbus have implemented Fact-GC on an LTMS and found it to be quite successful.

2.3 Belief Maintenance Systems

Belief maintenance systems (BMSs) are truth maintenance systems that are able to reason with uncertainty. Most BMSs are truth maintenance systems augmented with probability measures. Many rely on the Dempster-Shafer theory of belief functions where belief in a proposition expresses the probability of its provability [Ramoni and Riva, 1993]. The ability to reason under uncertainty is necessary because, as previously discussed, deductive closure in real-world applications is next to impossible and because agents often make assumptions that are shown to be false when the agent gets new information.

This thesis is concerned with reasoning under uncertainty because a reader of news articles will acquire beliefs from agents in news articles. The system might use these beliefs to deduce other beliefs for the reader. Upon obtaining new information, the acquired beliefs could be proven false.

Any beliefs deduced from the false beliefs will need to be revised or retracted. Furthermore, it would be absurd to assume deductive closure over the model. The model of a reader's beliefs should not be deductively closed since it implies that the reader knows (or believes) all the consequences of his beliefs.

Another source of uncertainty is in the degree of reliability of the belief or the degree of belief in the belief. Degrees of belief are notoriously difficult to express. However, they do exist. People are known to use modifiers with the concept of belief such as "a firm belief", "I strongly believe that...", or "I'm not sure I believe ..." which indicates doubt in the belief.

One belief maintenance framework that will be looked at here is Ballim's ViewFinder [Ballim, 1992]. ViewFinder is a framework that enables belief maintenance for the nested beliefs of agents. One very important contribution of this framework is in its method of representing beliefs which allows for the nesting of beliefs. This framework is well suited to the problem this thesis is attempting to solve: *how can a system simulate how a reader acquires beliefs from news articles to build a model of the reader's beliefs?* Firstly, beliefs extracted from news articles are necessarily nested beliefs. They are the reader's beliefs about the reporter's beliefs about the source's beliefs, etc. ViewFinder is the only framework that even considers the possibility of nested beliefs.

Secondly, ViewFinder introduces the concept of a viewpoint. John's beliefs about geography are beliefs from John's viewpoint. When considering nested beliefs, beliefs are seen from another agent's viewpoint i.e., the reader's view of the reporter's viewpoint.

Thirdly, ViewFinder introduces the concept of a topic environment. A topic environment contains all the beliefs of an agent about a particular topic. Given today's wealth of information, a reader can potentially have thousands upon thousands of beliefs. Topic environments serve to categorise and organise the beliefs. In addition, topic environments also serve to limit the scope within which a contradiction can occur. For example, John's topic environment for dogs can contain the belief that dogs are good while the topic environment for dogs in John's view of Peter's viewpoint can contain the belief that dogs are bad. However, neither of the topic environments can contain both beliefs. This is an important point since it greatly reduces the search space and makes detection of contradictions simpler.

ViewFinder has been chosen as the basis for the solution proposed in this thesis. The following section will provide a brief overview of ViewFinder describing some of its more important contributions. Chapters 4 and 5 will examine in detail the contributions of the framework that are most relevant to this thesis.

2.3.1 ViewFinder

In his dissertation [Ballim, 1992], Ballim proposes a framework, called ViewFinder, for representing, ascribing and maintaining the nested beliefs of interacting agents. Agents are defined as *people and intelligent programs interacting with each other*. The mechanism that generates the nested models can not do so deductively. Since the system or agents can not *know* what another agent believes, they can only make educated guesses. Therefore, the generation of the nested beliefs depends on making assumptions about the agents. Being assumptions, they are subject to belief revision if new information shows them to be false.

Ballim contends that a framework for representing, ascribing and maintaining nested beliefs must do the following:

- provide a representation medium for the system that is easy to understand. All information that the system needs to fulfill its functions must be represented and in a form that permits efficient reasoning over the information.
- allow for the attribution of propositional attitudes to agents based on the content of the propositions.
- permit retraction or revision of beliefs (attributed or otherwise) when they are proven to be false.

Belief Representation

The method by which nested beliefs are modelled must allow the representation and use of the following:

- the system's beliefs
- the beliefs of each agent with whom the system interacts
- the beliefs of agents about other agents and the system
- the system's beliefs about the goals and plans of the other agents
- the system's beliefs about the other agents' beliefs about their goals and plans, etc.

All of this representation must be done dynamically and allow for revision of the beliefs. Ballim introduces the concept of a *belief diagram* which is a tool for the representation of nested beliefs. Belief diagrams are based on earlier work by Ballim and others [Wilks and Bien, 1983, Wilks and Ballim, 1987, Ballim *et al.*, 1991, Ballim and Wilks, 1991] and fulfill all the requirements listed above. The representation of the nested beliefs is an important issue and will be discussed further in Chapter 4.

Belief Attribution

The dynamic transfer of beliefs from one view to another is called *belief attribution*. This is an important notion because system designers cannot know and be able to represent the beliefs of all the agents with whom the system will be interacting [Ballim, 1992] at the time the system is being developed. Systems working in complex domains must be flexible and able to integrate new information about agents and their beliefs. There are three types of belief attribution. The first is belief interpretation, the second is dynamic belief ascription, and the third is belief percolation.

Belief interpretation is the interpretation of an agent's utterances and actions resulting in beliefs about the agent. The simplest form of belief interpretation is explicit belief interpretation which consists of the interpretation of belief sentences such as "Mary believes that John likes her." On the other hand, implicit belief interpretation is the interpretation of normal utterances to determine underlying beliefs. Explicit belief interpretation can be very useful for attributing goals and plans to agents as they will sometimes directly state their goals when communicating with a system [Ballim, 1992]. Sidner and Israel [Sidner and Israel, 1981] propose two rules for attributing beliefs based on belief interpretation.

1. Sincerity Rule

   If x wants y to believe that x believes q, then x believes that q

2. Reliability Rule

   If y believes that x believes that q and that x is reliably informed about q, then y will believe that q

However, in real-life situations, people usually use a less direct approach to revealing their goals and other methods of interpreting beliefs are needed. Researchers Isozaki and Katsuno [Isozaki and Katsuno, 1996] propose a method for estimating the beliefs of other's based on observation of their actions.

Belief interpretation implies a direct personal knowledge of the utterances and actions made by an agent. Belief interpretation is not of concern in this thesis since readers of news articles are working with *reported* utterances and actions.

The second method of belief attribution is ***belief ascription***. Belief ascription is the dynamic transfer of beliefs from one view to another based not on personal observation of the other agent but on a basic principle - it can be assumed that other agents believe what we believe. This principle leads to what has been called the ***default ascription rule*** first introduced by Wilks and Bien. It states that: "X's view of Y can be assumed to be one's own view of Y, ***except where there is explicit evidence to the contrary***" [Wilks and Bien, 1983, p.103]. Belief ascription is discussed in detail in Chapter 5.

One side-effect of belief ascription as defined by Ballim is ***belief percolation***. Belief percolation occurs when one agent "learns" from another agent. For example, the system's view of John's view of Peter's viewpoint about Mary could include the belief that Mary is an engineer. If the system does not have this belief but has sufficient confidence in Peter's expertise on Mary, then Peter's belief could be percolated up to the system's viewpoint and become one of the system's beliefs about Mary. In this way, the system has learned from Peter. Belief percolation is a mechanism that is central to this thesis. It will be further discussed in Section 5.4. However, in order to adequately simulate how a reader could acquire beliefs from reading news articles, belief percolation needs to be made into an independent mechanism. Chapter 7 explains the modifications that have been made to Ballim's version of belief percolation to adapt it to the needs of this thesis.

Belief Revision

In the ViewFinder framework, belief revision is handled in the same manner as most truth maintenance and belief maintenance systems. ViewFinder defines operations on environments (viewpoints) that allow for reasoning with *preference schemes*. Preferences make explicit the different levels of commitment that an agent might have towards a belief through some sort of ordering of the beliefs. In addition, ViewFinder allows for partial orderings by making environments to be ***equivalence classes*** under a particular ordering relationship. ViewFinder proposes three belief revision subprocesses:

- simulating revision within an agent's belief model. This is akin to applying traditional belief revision within an environment.
- propagating revisions in one belief model through the chain of viewpoints
- changing the entire basis of a model of an agent's beliefs. This can occur if a set of beliefs has been attributed to an agent based on a stereotype and it is discovered that the agent is not a member of the stereotype.

Being a framework, ViewFinder allows for various belief revision strategies to be used without committing itself to one in particular.

In the case of the models of readers of news articles, belief revision occurs when a reader is presented with new information that is contrary to one of his beliefs. Belief percolation as redefined by this thesis allows for beliefs to be percolated into the reader's viewpoint as ***potential beliefs***. These beliefs may be contradictions of the reader's ***held beliefs***.

A second mechanism introduced in this thesis, ***belief promotion***, in combination with a ***belief heuristic*** allows a potential belief to become a held belief when the reader has acquired sufficient confidence in its reliability. If a contradiction in the held beliefs is detected, the new held belief is retained and the old held belief is removed. At this point, the second belief revision process described above would need to be invoked to propagate the revision through the viewpoints. However, since nested viewpoints are easily generated by the framework, it might be sufficient to simply regenerate the affected viewpoints.

Belief promotion and the guiding belief heuristic are described in Sections 6.4.6 and 6.5.4. ***Percolator***, the proof-of-concept system of the ideas introduced in this thesis, implements belief percolation and belief promotion for the acquisition of beliefs by a reader of news articles. Details of the implementation of the belief percolation mechanism and the belief promotion mechanism are described in Sections 7.3 and 7.5 respectively.

Chapter 3

Evidential Analysis of Reported Speech

The concern of this thesis is to enhance the ability of agents and information retrieval systems (IRSs) to determine which bits of information in the vast databases of on-line newspaper articles to present to the user. To do this, the agents and IRSs must be able to obtain clues from the semantic and syntactic structures of the texts. One important element of newspaper articles is the use of reported speech. North American newspapers use reported speech extensively to provide evidence for a proposition as opposed to indicating commitment to the embedded statements. It is left to the reader to evaluate the provided evidence.

Bergler [Bergler, 1992] has studied the use of reported speech in newspaper articles extensively and has developed an *evidential analysis* of reported speech. Evidential analysis provides a way of preprocessing a news article to extract the parts of the reported speech as well as a way of representing the article called profiling. The resultant profiles can then be used by a system such as the one proposed in this thesis as the basis for the beliefs that will be used to enrich the model of the reader. Although not all of the concepts described in this chapter have been automated as of yet, sufficient detail is provided here to show that full automation is possible and simulation of belief acquisition by readers of news articles is feasible. The following sections are a summary of Bergler's work on this subject as relates to the problem addressed by this thesis.

In reported speech, the embedded clause contains the *primary information*. This is the information that contributes to the main point of the news story. The matrix clause of the reported speech is the *evaluative environment* for the primary information. The criterion for the evaluation is *reliability*. When conducting an evidential analysis, the reader is trying to establish the reliability of the information being presented by the reporter. Bergler shows that the evaluation of reliability is conventionalised in *newspaper style* and that the evaluative environment is provided largely by the lexical semantics of the two functional parts of the matrix clause of the reported speech: the source and the reporting verb.

US Advising Third Parties on Hostages[1]

(R1) *The Bush administration continued to insist yesterday that* (C1) it is not involved in negotiations over the Western hostages in Lebanon, (R2) *but acknowledged that* (C2) US officials have provided advice to and have been kept informed by "people at all levels" who are holding such talks.

(C3) "There's a lot happening, and I don't want to be discouraging," (R3) *Marlin Fitzwater, the president's spokesman, told reporters.* (R4) *But Fitzwater stressed that* (C4) he was not trying to fuel speculation about any impending release, (R5) *and said* (C5) there was "no reason to believe" the situation had changed.

(A1) Nevertheless, it appears that it has. ...

Figure 1: Boston Globe, March 6, 1990

3.1 Structure of Reported Speech

This section will introduce some basic concepts about the structure of reported speech relative to syntax and semantics.

3.1.1 Syntax of Reported Speech

A newspaper article provides two levels of information. The first is the *primary information*. This is usually provided by an expert or witness being interviewed and is the information needed to push the story along. The second is the *circumstantial information*. This is usually provided by the reporter and gives details about the circumstances surrounding the primary information. It allows the reader to view the primary information in terms of a particular perspective or belief context. In newspaper articles, these two levels of information are distinguished syntactically. The matrix clause contains the circumstantial information (who, when and how) of the reported event. The complement clause contains the primary information (what) of the reported event. In Figure 1 from [Bergler, 1992], the circumstantial information is shown in italics.

The reporting verb chosen by the reporter contributes information about the manner in which the primary information was stated, the preciseness of the quote, as well as the temporal relationship between the primary information and the circumstantial information.

Reported speech can come in many different syntactic structures. However, the structure, in newspaper articles, is often spread over one or more sentences. The various parts of the utterance can be direct quotes or paraphrases. In addition, the position of the source and reporting verb vary. They can be placed before the utterance, in the middle of the utterance or at the end of the utterance. The source can come before or after the reporting verb. A few of the more typical structures are listed below[2].

1. <Source> <reporting verb> "<utterance>"

2. "<Utterance>," <reporting verb> <source>

[1]Figure 4.1 in [Bergler, 1992].
[2]Structures 1, 4, 5 and 8 are from Example 2 [Bergler, 1992, p.42].

3. "<Utterance>," <source> <reporting verb>

4. "<Part of utterance>," <source> <reporting verb>, "<rest of utterance>"

5. "<Part of utterance>, " <reporting verb> <source>, "<rest of utterance>"

6. <Source> <reporting verb>,"<Part of utterance>" <paraphrase of rest of utterance>

7. "<Part of utterance>," <source> <reporting verb>, <paraphrase of rest of utterance>

8. <Source> <reporting verb> (that) <paraphrase of utterance>

A quick perusal of the Seattle Post-Intelligencer[3] provides numerous examples (see below) of these syntactic structures.

1. <Secretary of State Madeleine Albright>, also travelling with Clinton, <said> "<We have made very clear that were there any attacks on our forces or on neighboring countries that our response would be swift and sure.>"

2. "<He's picking winners and losers, and he's trying to give something to everybody in the room>," <said> <Knollenberg, R-Mich.>

3. "<She would be terrific>," <Clinton> <said.>

4. "<In Turkey>," <Ocalan> <insisted>, "<the only avenue open to the Kurds is to take up arms.>"

5. "<We were not looking to see what was best for AT&T or for Internet service providers>," <said> <Councilwoman Tina Podlodowski>, "<but what was best for the customer.>"

6. Asked whether the first lady might run for the Senate <Clinton> <said>,"<She'd be great if she did it>", <but has not had enough time to think about it.>

7. "<We need to start turning this situation around>," <Samet> <said>, <adding that the legislation would help drivers deal with youths bringing guns on buses and brandishing them to intimidate the people on board.>

8. After a Paris meeting of European Union foreign ministers, <German Foreign Minister Joschka Fischer> <said> that <all members of the union had supported military enforcement of the peace accord.>

3.1.2 Semantics of Reported Speech

Reported speech is guided by the notion that the reporter is trying to inform. The reporting verb serves to attribute an utterance to a source. By doing so, the reporter makes the commitment to be faithful to the original utterance. To do so, the reported utterance needs to be an accurate interpretation of the original utterance in its original context. And, when the current context of the

[3] Seattle Post-Intelligencer, February 16 and 17, 1999

utterance is very different from the original context, some indication of the original context should be given. As previously stated, this ***circumstantial information*** is usually provided to the reader in the matrix clause. According to [Bergler, 1992, p.42], the basic meaning of a reporting verb can then be characterised as follows:

(1) $paraphrase-of(B_{org}, B)\ \&\ utter(A, B_{org})\ \&\ utter(C,\ utter(A, B))$
where
A is the source,
B is the utterance,
B_{org} is the original utterance, and
C is the reporter

Readers assume that the reporter actually witnessed the source making the original utterance. To allow readers to make this assumption when this is not the case, would be a breach of journalistic integrity. So, it is the reporter's duty to indicate to the reader that the information was obtained indirectly unless the context of the reported speech itself overrides the default assumption. [Bergler, 1992, p.43] rewrites Definition (1) to incorporate this default assumption as follows:

(2) $paraphrase\_of(B_{org}, B)\ \&\ utter(A, B_{org})\ \&\ utter(C,\ utter(A, B))$
$\&\ default:\ witness(C, utter(A, B_{org}))$

3.2 Evaluation of Reported Speech

When a reporter uses reported speech to attribute material to a source, he is not committing to the veracity of that material. The reporter may not even believe what is being reported. The reporter should be truthful in his reporting and should try to characterise the situation or context in which the original speech occurred. The reporter should try to include any relevant details of the original situation to enable the reader to perform his own interpretation of the original speech.

The goal of any reader of news articles is to gain knowledge. We must consider two stereotypical classes of readers. There is the naive reader who believes everything that he reads. This type of reader does not interpret the original speech situation. Rather, he relies on the reporter's encoded evaluation and makes the assumption that all the information is reliable. The other class is that of the more informed reader. The informed reader uses his own background knowledge to interpret the original speech situation. This background knowledge includes such things as the reader's own assumptions about the reporter's and source's goals and interests, the biases of the newspaper or web site as an institution. The informed reader may also have assumptions about the opinions, beliefs, biases and points of view of the reporter. Therefore, the tasks the reader must accomplish when reading a news article are (i) to interpret the article within the viewpoint of the reporter and, (ii) to assess his own beliefs and points of view and to compare those to the assumed points of view of the reporter and the newspaper (or news web site). In the terminology of Ballim and Wilks [Wilks and Ballim, 1987, Ballim and Wilks, 1992, Ballim, 1992], the reader would *ascribe* beliefs into the *belief environments* of the reporter and the newspaper. Then, he would attempt to *percolate* beliefs into his own environment (see Chapter 5.4 for a discussion of belief percolation).

Interpreting the article from within the viewpoint of the reporter involves two sub-tasks implied by Definition (2) above. First, the reader must identify the situation of the original utterance, the *original context*, by collecting all the circumstantial information. The reader then is able to evaluate the reported speech in the context of the original utterance. The reader must then establish the function of the report. The reported speech must be evaluated in its overall context: that of the news article. This is called the *current context*.

Bergler contends that the task of identifying and evaluating the original utterance must occur before that of determining the function of the reported speech. The basis for this is that "the reporting clause provides a partial description of the original utterance situation, providing an *evaluative* environment. Within this evaluative environment the reliability of the embedded statement can be determined." [Bergler, 1992, p.45]

The following subsections will explain how the lexicalisation of the source and the reporting verb can be used by the reporter to encode information about the original speech context. The reader can then make his own judgement on the reliability of the reported speech.

3.2.1 Lexical Semantics of the Source

The primary goal in the interpretation of the original utterance is to establish its reliability which depends directly on the assessment of the source. The reporter, if he so chooses, can impart much information about the source to help the reader to determine, first, the reliability of the source and, second, the reliability of the reported material. For instance, the source's identity, expertise, experience and relevance to the situation provide insight into the situation of the original utterance and help the reader to determine the reliability of the source. The reporter can also encode his own assessment of the source, thus giving the reader clues as to how the reporter wants the reader to evaluate the source and interpret the reported speech. This is done through the lexical realisation of the source. How the reporter refers to the source, indicates the manner in which the reader is to interpret the reported speech.

Bergler sees two main reasons for the trustworthiness of a source: inherent and incidental. A source is inherently trustworthy if he is an authority or expert on the subject of the reported speech. The authority or expert should be recognised by others in the domain. A source is said to have incidental trustworthiness if relevant evidence is available to him or if he has some involvement in the subject of the reported speech. A reporter might encode his opinion about a source's authority, expertise, access to evidence and involvement by calling the source an *official*, a *researcher*, a *witness* or a *victim*. Bergler points out that these categories are not mutually exclusive. For instance, a *specialist* while an expert in his field will also usually have good evidence. And, on the other hand, an otherwise trustworthy source may not have adequate knowledge about the reported material.

Establishing competency is an issue beyond the scope of this thesis but in the newspaper domain, there is a strong assumption on the part of the reader that the reporter has chosen competent sources.

1. an aircraft accident expert, Bernard Coogan
2. Mark Potok, the law center's spokesman
3. Dr. Tommy Brown, a pathologist,
4. Craig Johnson, owner of the World of Sounds music store
5. Commission Chairman Ann Brown
6. Gary Gardner, executive director of the Washington Association of Internet Service Providers,
7. the Pentagon's spokesman, Kenneth Bacon
8. Alan Greenspan, chairman of the Federal Reserve
9. Lord Levene, Lord Mayor and a leading figure in the City who has warned about the risks of staying out of the euro-zone
10. Matthew Barrett, chairman and chief executive of Bank of Montreal
11. Franco Bernabe, the Telecom Italia chief executive
12. Marks and Spencer, the UK's biggest clothing retailer
13. a company source
14. President Clinton's top foreign policy advisers
15. one lawyer close to the case
16. Agency officials
17. officials at the Pentagon

Figure 2: Examples of NP Source Lexicalisations

The reporter usually indicates his opinion on the competency of the source through his choice of lexical realisation of the source, called source noun phrases (or source NPs). As can be seen from the examples[4] in Figure 2, reporters often include along with the source's name some information about the source's ***authority***. The reporter is trying to establish the source's credibility (or lack thereof) by giving the reader some circumstantial information, i.e. the original context. Example 12 is a case where the source is actually a company. And, in the cases of examples 13 to 17, the reporter didn't include the name of the source (perhaps due to the source's desire for anonymity). However, the reporter chose formal sounding lexicalisations to justify the source's authority (example 17), expertise (example 14), evidence (example 13) or involvement (example 15).

[4]Examples from USA Today (usatoday.com),the Financial Times (ft.com) and the Seattle Post-Intelligencer (seattle-pi.com) February, 1999.

vocalisation	neutral	contextual	pragmatic intent
cry	say	allude	accuse
call	tell	argue	admit
mumble	report	contend	claim
mutter	relate	insist	deny
scream	announce	reiterate	promise
shout	release	reply	joke
stammer	state	dispute	assure
stutter	add	ask	pledge

Table 2: Reporting Verbs Encoding the Manner of the Original Utterance

3.2.2 Lexical Semantics of the Reporting Verb

The lexical semantics of the reporting verb plays an important role in the task of interpreting the original utterance. The reporting verb partly specifies the situation of the original speech utterance, the original context. It can be used to partly specify the manner, intention, and force of the original utterance. Bergler [Bergler, 1992] studied the field and identified how reporting verbs can be used to reconstruct aspects of the original context. Table 2[5] shows how reporting verbs can encode the *manner* of the original utterance. *Neutral* reporting verbs do not specify the manner of the original utterance. *Vocalisation* verbs specify the physical aspects of the original utterance such as high or low voice, clear or unclear voice. *Contextual* verbs indicate coherence of the original utterance in its original context. Such verbs would, for example, indicate whether the original utterance was a reiteration of or a reference to a previously made point. *Pragmatic intent* is a little different in that it goes beyond the textual manner by specifying intentions with pragmatic ramifications.

Other classification dimensions could include such things as the textual status of the reported material within the context of the original utterance. Textual status can be used to differentiate new and previously mentioned or implied material. Previously mentioned or implied material can be further broken down into confirming material and contrasting material. Manner and textual status are but two of many possible semantic dimensions in the field of reporting verbs.

Bergler [Bergler, 1992, Bergler, 1993] thoroughly analysed the field of reporting verbs and has identified a set of semantic dimensions that can be used to differentiate reporting verbs. Section 3.4 will give an overview of that analysis and will explain how the semantic dimensions of the field of reporting verbs can be used in the evaluation of the reliability of the reported speech.

3.2.3 Evaluative Context of the Reporting Clause

Sections 3.2.1 and 3.2.2 showed how the lexical semantics of the source and of the reporting verb can be used in the interpretation and evaluation of the original utterance in the original context. This is, in fact, the first subtask involved in interpreting the article from within the viewpoint of the reporter. The second subtask is to establish the function of the report. The reported speech must be evaluated in the overall context of the news article: its *current context*. The function of

[5]Figure 3.4 in [Bergler, 1992].

the evaluative context of the reporting clause is to determine whether the propositional content of the reported clause is ***primary information*** or ***ancillary information***. As discussed in section 3.1.1, primary information is information that moves the story forward whereas ancillary information is information that serves as support for points previously made in the story. It is important to know what type of information is being dealt with in order to correctly direct the reasoning process.

3.3 Evidential Analysis of Reported Speech

Bergler's evidential analysis of reported speech in news articles assumes that primary information is contained in the reported clause. This is information that the reporter wants the reader to know but which can also have a degree of unreliability since the reporter has neither first hand knowledge of the reported material nor has expertise pertaining to the reported material. The reporter attributes the information to a source. However, the reported speech still encodes the reporter's evaluation and opinions as to the material being reported. The encoding occurs through the overall argument presented as well as through the choice of reporting verb and the description of the source, as discussed earlier.

Evidential analysis allows the reader to judge the validity of the complement clause based on the reliability of the evidential scope. Evidential scope consists of an interpretation of the reported clause given a set of context variables (original context, current context and temporal context). This is crucial as it means that the complement clause can be evaluated in isolation as would be necessary in the case of keyword based searches. It can be evaluated as needed in a localised context stored in the context variables which indicate its reliability.

(3) $SAY(Source, \theta\prime)$ *is true* $\Rightarrow \theta\prime[OC, CC, TC]$[6]
where:
OC is the original context
CC is the current context
TC is the temporal context

Evidential analysis contends that the complement can be considered true if the circumstantial information is determined to be reliable. Bergler advances that the practical implications of storing the context in so called context variables is that reliability can be determined on demand and can take into consideration the user's particular set of beliefs and points of view. The following will walk through an extended example, taken from [Bergler, 1992, p.61], of an evidential analysis of the text in Figure 1 that will yield some evaluation of the reliability of the reported speech. The primary information in the text can be represented in the following format which incorporates evidential scope (assume that C3 and C4 have been identified as ancillary information and are represented separately):

BA = "the Bush administration"
E1 = λx: NEGOTIATE(x, RELEASE(CAPTORS, "the Western hostages in Lebanon"))

[6]Bergler [Bergler, 1992, p.61]

C1	=	NOT(E1(BA))
R1	=	(OC: (S: BA; U: REPEAT(INSIST); C: YESTERDAY), CC: NIL, TC: PAST-TENSE(R1), PRESENT-TENSE(C1))
S1	=	C1(R1)
USO	=	"US officials"
PL	=	"people at all levels" & E1(PL)
C2	=	ADVISE(USO, PL) & INFORM(PL, USO)
R2	=	(OC: (S: BA; U: ACKNOWLEDGE), CC: S1 "but" S2, TC: PAST-TENSE(R2), PRESENT-PERFECT(C2))
S2	=	C2(R2)
MF	=	"Marlin Fitzwater"
E2	=	λu: BELIEVE(u,CHANGED(SITUATION))
C5	=	$\forall$y: NOT($\exists$z: REASON(z, E2(y)))
R5	=	(OC: (S: MF; U: SAY), CC: S4 "and" S5, TC: PAST-TENSE(R5), PAST-TENSE(C5))
S5	=	C5(R5)

The variables S, U, and C in the original context variable OC represent the source, the reporting verb and additional information about the original context. Evidential analysis provides a way of indexing the information in a text based on its function, i.e. source, primary information, etc. Now it would be possible to retrieve articles based on the following query: "articles that show U.S. government involvement in negotiations over Western hostages in Lebanon in March 1990"[7]. The approach to be taken to this query is to restrict the search to articles from March 1990 and to articles concerning Western hostages in Lebanon. Analyse the complements (C-clauses) for synonyms of "negotiations" and "possible release". For every statement retrieved, evaluate its reliability by determining the reader's confidence in the source and the reliability encoded in the reporting verb. If the reliability satisfies some criteria, retrieve the article. Bergler proposes the following procedure:

$\forall$ articles do

if date $\subset$ March 1990

[7]Scenario 3 in Bergler [Bergler, 1992, p.66]

then ∀ i do
  if topic of Ci = U.S. government involvement in negotiations over Western hostages in Lebanon
  then **if** RELIABILITY(Ri) = high
    then retrieve article

RELIABILITY for the first two sentences of the text in Figure 1 is represented as follows:

RELIABILITY(R1)
  COMBINE
    RELIABILITY(R1:OC:S) < mass term, no responsible source specified >
    RELIABILITY(R1:OC:U) < *insist*, presupposes opposition >
    RELIABILITY(R1:OC:C) < no filler >

RELIABILITY(R2)
  COMBINE
    RELIABILITY(R2:OC:S) < mass term, no responsible source specified >
    RELIABILITY(R2:OC:U) < *acknowledge*, factive verb >
    RELIABILITY(R2:OC:C) < no filler >

Based on additional evaluation criteria, not provided in Bergler's example, an intelligent system could determine that R2 has a high enough reliability to retrieve the article because what is being *acknowledged* is usually not in favour of the source. It is assumed that one would not acknowledge something bad about oneself unless it were true. Of course, the system would also need to be able to equate the notion of "negotiation" with that of "advising and informing" which are the contents of C2.

3.4 Semantic Dimensions

This thesis is concerned with the semantic dimensions of reporting verbs which are used to report the speech of others. They are much used in North American newpapers since most statements are supported by a source quoted directly or indirectly using a reporting verb. The verb used by the reporter provides an evaluation of the reporter's belief in the credibility of the information. However, in order to correctly understand the contribution of reporting verbs, each reporting verb must be compared with others in the same semantic field, i.e., other reporting verbs.

[Bergler, 1993] has defined "a set of semantic notions that distinguish the members of the field from each other" which she calls semantic dimensions. She has defined nine such semantic dimensions for the field of reporting verbs. Many more could have been defined to distinguish every instance of a reporting verb but Bergler has made a judgement as to the desired granularity needed in this field and contends that the nine defined semantic dimensions are sufficient for the purpose of evidential analysis. The semantic dimensions are divided into three categories. The dimensions in the first category define the situation of the original utterance, i.e. the original context. The dimensions of the second category define attitudes toward the complement clause and the third category defines the strength of the complement. Table 3[8] shows a summary of the possible values of the nine semantic dimensions in the field of reporting verbs.

Situation of the Original Utterance

Voice Quality specifies the quality of the source's voice at the time of the original utterance. The possible values to describe voice quality include the entire spectrum of voice descriptions such as high - low, clear - unclear, high pitched - low pitched, etc. The default is ***unmarked***.

Explicitness The possible values are explicit and implicit. Although there are many possibilities in between such as ***to hint***, ***to suggest***, etc. Bergler contends that such a scaling of explicitness would not be useful. The default is ***explicit***.

Formality The possible values are formal and informal. The default is ***unmarked***.

Audience The possible values are public and private. The default is ***unmarked***.

Attitudes toward the Complement Clause

Polarity This is used to indicate whether the source is asserting the complement or its contrary. The possible values are positive and negative. The default is ***positive***.

Presupposition This indicates whether the reported clause contained new or presupposed information in the context of the original utterance. The default is ***unmarked***.

Speech Act The values for this dimension can be any of the generally accepted speech acts such as inform, request, question, etc. The default is ***inform***.

Affectedness This dimension refers to the impact of the reporting clause on the source. The possible values are positive and negative. The default value is ***unmarked***.

Strength of the Complement

Strength This is used by the reporter to indicate whether or not there is room for doubt as to the reliability, certainty or credibility of the complement clause. The possible values are high and low. The default is ***unmarked***.

The semantic dimensions can be divided into two classes: essential and optional. The essential dimensions are those that have "unmarked" as the default. These dimensions will always be present. Usually, reporting verbs will only encode one optional dimension. The result of this is that the optional dimension becomes the primary characteristic of the reporting verb. Table 4[9] gives an example of the semantic dimensions for some reporting verbs. As can be seen, ***thunder*** and ***say*** are distinguished by the voice quality dimension. Since voice quality is an essential dimension and is

[8]Figure 4 in Bergler [Bergler, 1993]
[9]Figure 5 in Bergler [Bergler, 1993]

Course Structure	Semantic Dimension	Values	Default
Situation of the Original Utterance	voice quality explicitness formality audience	range: high,low,clear,... explicit - implicit formal - informal public - private	unmarked explicit unmarked unmarked
Attitude to Complement	polarity presupposition speech act affectedness	positive - negative new - presupposed range: request, question,... positive - negative	positive unmarked inform unmarked
Strength of Complement	strength	high - low	unmarked

Table 3: The Semantic Field of Reporting Verbs

the only difference between *say* and *thunder*, the two verbs could be considered synonymous in an appropriate context. On the other hand, *concede* and *deny* are distinguished by the optional dimension polarity. The dimension forces a distinction between positive polarity and negative polarity. As such, *concede* and *deny* can never be considered synonymous.

Verb	Voice	Explicit	Formal	Audience	Polarity	Presup.	Speech Act	Affect.	Strength
thunder	loud	explicit	-	-	pos.	-	inform	-	-
say	-	explicit	-	-	pos.	-	inform	-	-
claim	-	explicit	-	-	pos.	-	inform	-	low
affirm	-	explicit	-	-	pos.	presup.	inform	-	high
concede	-	explicit	-	-	pos.	presup.	inform	neg.	-
deny	-	explicit	-	-	neg.	presup.	inform	neg.	-
allude	-	implicit	-	-	pos.	new	inform	-	-
announce	-	explicit	-	public	pos.	new	announce	-	high
declare	-	explicit	formal	public	pos.	new	announce	-	high

Table 4: Semantic Dimensions of Some Reporting Verbs

3.4.1 Semantic Dictionary Entries

The intent behind the analysis of the semantic dimensions of reporting verbs was to devise a template that could be used to encode the "meaning" of the verbs. Based on some earlier work with the *Generative Lexicon* [Bergler, 1995b, Pustejovsky, 1991], Bergler created a template that when completed could be used to encode the meanings of verbs in the field of reporting verbs which together with a template for other word categories could consitute a semantic dictionary.

Figure 3 shows the entry for the reporting verb *to insist* using the template for reporting verbs developed by Bergler.

CLASS refers to the type of entry, in this case the class of reporting verbs. ROOT is the base form of the verb. POS is the part of speech. SENSE is used to indicate which meaning of the verb is being documented. ARGS is the number of arguments that the verb takes. SUBJ is the subject of the verb. AGENT is the default subject. REQ indicates whether or not the subject is required. SUBCAT states the subcategorisation of the subject. The default is that the subject of the verb must be human. MET-EXTENSIONS (metonymic extensions) refers to the fact that the source in news articles is often an entity other than a person such as a document or an institution. This slot in the template indicates which if any such entities can be used as the subject of the reporting verb.

```
CLASS <rv>
ROOT insist
FORMS insists insisted insisting
POS verb
SENSE 1
 ARGS 2
  SUBJ default AGENT
   REQ default yes
   SUBCAT default human
   MET-EXTENSIONS document, company, institution
  OBJ1 SCOMP
   PREP. optional default that scomp ''that''
   REQ yes
   SUBCAT information
   MET-EXTENSIONS
  OBJ2
   PREP.
   REQ
   SUBCAT
   MET-EXTENSIONS
 LCP
 EVENT-TYPE transition
 TEMPORAL
 DEFINITION to state, emphasise or hold firmly to
 EXAMPLES
 HYPERNYMS
 SYNONYMS
 ANTONYMS
 SEMANTIC-CONCEPT RV
 LC
  VOICE default unmarked
  EXPLICIT default explicit
  FORMAL default unmarked
  AUDIENCE default unmarked
  POLARITY default positive
  PRESUPPOSITIONS presupposed
  SPEECH-ACT default inform
  AFFECTEDNESS default unmarked
  STRENGTH default unmarked
 DISCOURSE-POLARITY
```

Figure 3: Semantic Dictionary Entry for the Verb ***to insist***

OBJ1 is the first object of the verb. In the case of *to insist*, it is the second argument (the subject is the first). Most reporting verbs have an SCOMP as the first object. If the object can take a preposition, it is listed after PREP. In this case, the object is a sentential complement (SCOMP) which takes an optional "that" preposition. OBJ2 is used if the verb has three arguments. LCP is the lexical conceptual paradigm. The EVENT-TYPE is by default that of a *transition*. The semantic dictionary should at least be able to distinguish states, transitions, processes, accomplishments, and achievements. TEMPORAL is the temporal information such as the tense of the verb. The SEMANTIC-CONCEPT in this case is that of a reporting verb.

The section of the template refered to as LC, the lexical concept, contains the possible semantic dimensions with the default values. When completing particular entries, only the non-default dimensions need to be mentioned. The nine semantic dimensions are described in Section 3.4. DISCOURSE-POLARITY is similar to the semantic dimension polarity. It indicates whether the verb has negative or positive polarity with respect to the discourse.

A semantic dictionary could be used by a system such as the one proposed in this thesis in several ways. First, the verb entries would be used when performing evidential analysis of the reporting verb to determine the reporter's confidence in the reported material. Secondly, noun entries could be used to give the system some knowledge of the semantics of the beliefs. The system could determine the reader's confidence in the source or the reporter by analysing the reader's beliefs. For example, if the reader has the belief unethical(reporter), using the semantic dictionary, the system could determine that the reader has low confidence in the reporter. A semantic dictionary gives a system some insight into the semantics or *meaning* of the beliefs. Without it, beliefs are meaningless black boxes being manipulated by a system.

3.5 Profiling

A method for representing text developed by Bergler is called profiling [Bergler, 1995a, Witte, 1994]. A profile is a collection of information pertaining to one specific source in a news article. It can contain such things as the attributes and features of the source, expressed thoughts and beliefs attributed to the source and all utterances made by the source. Having all the information about a source gathered in one location is useful for many tasks such as summarizing, information retrieval or in-depth discourse analysis. When a source is part-of or a member-of another source, such as an employee being part-of a company, we have a situation of embedded profiles. Bergler has chosen to illustrate profiles using boxes in a manner similar to the way in which Ballim and Wilks [Ballim and Wilks, 1992] represent nested beliefs. Since a profile is created for each source (agent) mentioned in the article, and since certain embedded relationships are shown as nestings, profiles can serve as a starting point for generating viewpoints needed to represent nested beliefs. In general, an entire text can be represented in a complex profile structure.

How can one go about building a complex profile structure for an article? Bergler points out several heuristics. First, often the first sentence of an article will consist of a summary of the main points of the article and usually introduces the people, institutions, etc. that are the topic of the article. The principal or *root source* is most often the first source introduced in the article. The identification of the root source is crucial when constructing complex profile structures because other sources are viewed as confirming or contradicting the root source. Second, any conflict of opinion or interest is usually mentioned in the first sentence of the text. This is very important when determining the supporting group structure.

When writing an article, the reporter has a particular goal or purpose in mind. He intends to support or oppose a particular topic or event. As mentioned earlier, there is a root source. The root source will be the principal source for the information supporting the reporter's intent. For example, if the reporter's goal in writing a particular article is to show support for military action, the reporter will select a source who is also in agreement. Reporters will often then cite other sources in argreement with the root source to further bolster their claims or to elaborate on the topic. All of these statements can be grouped together in what is called *supporting groups*.

There are many reporters who favour a more argumentative style or who are simply covering events and situations such as interviews and court proceedings where the discourse is a stylised argument. Participants have well established stereo-typed roles and these roles form associated supporting groups. Furthermore, there is an underlying relation between the utterances made by members of the different supporting groups, namely, contradiction. Many *argumentative* articles will feature statements by several sources on both sides of the story, these groups are called *opposing supporting groups*. Such knowledge of different types of discourse can be very useful when building profiles for text representation. The following, borrowed from Bergler [Bergler, 1992][10], is a representation of a newspaper article complex profile structure. The article contains three supporting groups: plaintiff, defendant, and committee. The plaintiff and defendant supporting groups are in fact opposing supporting groups and the committee supporting group seems to be in allegiance with

[10]Pages 181–183 in [Bergler, 1992].

the plaintiff supporting group.

Plaintiff

Regional thrift regulators from the San Francisco Office of Thrift Supervision

- charged that phone calls were bugged during the examination of Lincoln Savings & Loan Association
- charged that federal officials in Washington delayed Lincoln's seizure until its $2.5 billion cost made it the most expensive thrift failure ever.

William Black, acting district counsel for the San Francisco region of thrift regulators

- said "Clearly, we were shot in the back ... as we battled to protect the taxpayers."

- testified that a Big Eight accounting firm, Arthur Andersen & Co., participated in back-dating loan documents
- testified that the Washington officials even agreed in one document not to prosecute Lincoln over certain infractions.
- said Washington's chief thrift regulator, Danny Wall, and his principal lieutenants repeatedly ignored warnings that Lincoln was being operated in a reckless manner, certain to cause its failure.
- accused Mr. Wall of holding improper meetings with Lincoln officials, while refusing to listen to field examiners.
- said that Andersen backdated data to support loans that were made with no underwriting standards.

Mike Patricia, acting principal supervisory agent with the Office of Thrift Supervision

- said over two years, until April 1986, $1 billion in loans were approved, even though Lincoln had no written loan standards.
- said fifty-two loans were made in March 1986, and none had credit reports or other background work completed.
- said "At a later time, the files looked good because they had been stuffed"

Witnesses

- said the room in which California examiners were auditing Lincoln this past spring was bugged.

A government memo

- 'according to which' regulators tool control of the thrift in August discovered that a phone line in Lincoln's headquaters in Irvine, Calif., had been "compromised" to allow calls to be monitored on other extensions.
- didn't say who was monitoring the calls,
- said the matter was turned over to the Federal Bureau of Investigation.

Defendant

A Lincoln spokesman

- said its management "never authorized or participated in any bugging of anyone."

Mr. Wall

- is scheduled to testify Nov. 7

Mr. Wall's agency

- said that all sides of the issue should be heard before drawing conclusions

Other Washington thrift officials

- disputed the regional examiners' statement that they had called specifically for the seizure of Lincoln in 1987
- saying that it was only one option they presented.

Leonard Bickwit, a Washington attorney for Lincoln

- conceded that some memos had been written after the fact.
- said that "memorialization of underwriting activities that had been undertaken at an earlier time" did occur
- said that Lincoln believed it adhered to lending standards superior to the industry average.

A spokesman for Arthur Andersen

- denied any improprieties
- adding "At the request of our then-client, we provided staff personnel to work for a limited period of time under the direction of client personnel to assist them in organizing certain files."

Committee

Members of the Banking Committee

- sharply criticized their [the regional thrift regulator's -sb] Washington bosses, who had relieved them [the regional thrift regulator's -sb] of their responsibility for Lincoln two years before the thrift's parent filed for bankruptcy-law protection.

Committee Chairman Henry Gonzalez (D., Texas)

- said Mr. Wall "willingly cut the legs out from under his regulatory troops in the middle of the battle."
- renewed his call for Mr. Wall to step aside.

Rep. Joseph Kennedy (D., Mass.)

- said "The higher up in the regulatory process, the more corrupt it would appear."

The profiling of news articles is important because it separates some very important information that can be fed into a system like the one proposed in this thesis. It separates the information that is being reported from the circumstantial information. The reported information can then be translated into propositions that can be used by a system to create beliefs. It clearly identifies the sources and their lexical realisations which will be used to determine the reporter's confidence in the source. It also extracts the reporting verb from the sentences. The reporting verb is then used to evaluate the reporter's confidence in the reported information. This information will be used along with some other information by a belief heuristic which will determine the reliability of the beliefs. In addition, the structures represented in the complex profile structure can help a system to generate the nesting of viewpoints that will be needed to simulate how a reader acquires beliefs from news articles.

Chapter 4

Points of View in ViewFinder

Earlier work in artificial intelligence produced systems that could interact with other agents such as users or other systems. However, they did so at a very superficial level. Some could hold a simple "conversation" with an agent (i.e. Weizenbaum's Eliza) using pattern matching and some rules. In order to make the systems useful, researchers need to increase the complexity of the system's interactions. Systems may need to respond differently to different users depending on their knowledge of a particular field or topic. For example, a computer help or tutorial system would need to answer questions based on whether the user is new to computers or is a computer expert.

A system that needs to dialogue with other agents needs to "know" certain things about the other agents such as their goals, intentions, plans, and typical beliefs. Furthermore, the system needs to know what beliefs each agent has about the beliefs of other agents. For example, *Jack believes that Shannon believes that toads cause warts.* Jack's beliefs about Shannon are what is known as *Jack's point of view.* The meaning of the term *point of view* here must be distinguished from its use in [Wiebe, 1990]. Wiebe uses points of view to identify characters in a narrative in order to interpret a particular sentence, clause or paragraph from that character's point of view. In ViewFinder, a point of view, or viewpoint is a collection of all of an agent's beliefs. Jack's beliefs about Shannon are part of Jack's viewpoint and Mary's beliefs about Shannon are part of Mary's viewpoint. Although both contain beliefs about Shannon, they represent a different perspective and will most likely include different beliefs. The scope of what must be known or the levels of nesting of viewpoints depends on the purpose and type of dialogue or interaction. The work of Kaminka, Tambe and Hopper [Kaminka and Tambe, 1998, Kaminka *et al.*, 1998] demonstrates that agents (or systems) that can model the beliefs of other agents are much more robust. The manner or techniques by which a system acquires and represents this knowledge has developed into the field of belief modelling, also called user or agent modelling [Ballim and Wilks, 1992]. The modelling of agents in a multi-agent environment is a particularly difficult task which is well handled by the ViewFinder framework.

4.1 Belief Models

The first type of models that were developed focused primarily describing an object or concept. Minsky's work with frames [Minsky, 1975] was an early example of this. Each *frame* is a complex grouping of knowledge. It contains slots that enumerate the attributes of an object or concept. The slots are filled with new pieces of knowledge. This led to the development *scripts* by Shank and Abelson [Schank and Abelson, 1977]. A script is similar to a frame but consists of sequences of events that describe what needs to be done to accomplish an everyday activity. Later, scripts were augmented using structures to represent goals, plans and theme which describe actions in terms of a desired end. The Restaurant Script would contain a plan for what should be done at a restaurant to accomplish the goal of EATING: enter, wait to be seated, look at menu, wait for waitress, order food, wait for food, eat food, wait for waitress, pay for food.

As more work was done in the field, it became apparent that more sophisticated representations were needed. In an attempt to explain the phenomena of mutual belief (when two agents believe that they each have the same belief as the other) and deception (when one agent is lying to another agent about his beliefs) using models of agents' beliefs, Taylor and Whitehill [Taylor and Whitehill, 1981] proposed a representation that involved the nesting of models. Their representation uses cyclic structures to represent complex nestings. Given the sentence "Maggie tells Andy that she was once married", we can say that Andy's beliefs about Maggie (AM) are equal to Maggie's beliefs. We can also say that Maggie's beliefs about Andy (MA) are equal to Andy's beliefs. So AM=M and MA=A. Taylor and Whitehill contend that their representation using cyclic structures can capture all the information about the infinitely recursive model. Figure 4 is Taylor and Whitehill's representation of the two belief sets[1].

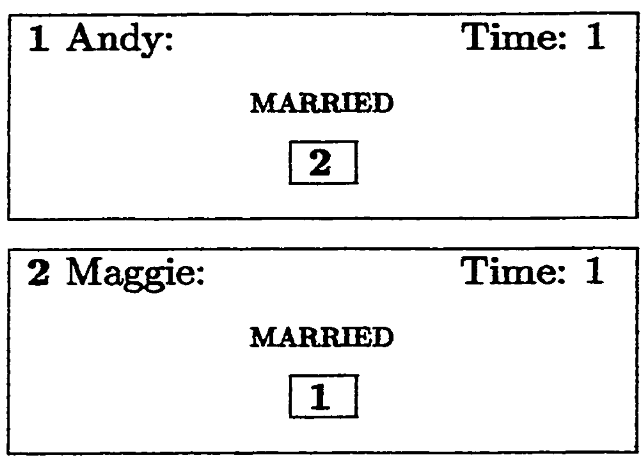

Figure 4: Nesting using Cyclic Structures

The structure represented in Figure 4 is one of mutual belief. The two large boxes with the number in the upper-left-hand corner are the two belief models. The models contain a list of

[1]Figure 1 in [Taylor and Whitehill, 1981]

facts (abbreviated in the figure but which would normally be a knowledge representation such as Conceptual Dependency) and one or more nested models represented by a smaller box containing a number that refers or points to an already existing belief model. In this example, MARRIED is the only fact listed in the two belief models. Box 1 represents Andy's belief that Maggie was married at Time: 1. The nested model points to box 2 (Maggie's beliefs). So this can be read as Andy's belief that Maggie believes... (the contents of box 2). Box 2 represents Maggie's belief that she was married at Time: 1. The nested model points to box 1 (Andy's beliefs). So, this can be read as Maggie's belief that Andy believes... (the contents of box 1). The cyclic nature of these two belief models can represent any belief of the form "Andy believes that Maggie believes that..." or "Maggie believes that Andy believes that..."

Taylor and Whitehill's representation scheme, although useful for representing mutual beliefs, lacks in many other areas. The most important of which is that it does not provide any mechanism for organizing the beliefs. Even as far back as 1975, Minsky [Minsky, 1975] argued in relation to his frame theory that knowledge and belief must be organised around topics. He further stated that it was his belief that human reasoners must have models of each other in order to reason. Ballim and Wilks have put these two ideas together and claim that "believers have models of each other and of other entities and that these models can be [...] *nested* within each other so as to produce *spaces within which to do reasoning*." [Ballim and Wilks, 1992, p.28] Based on this claim, they have come up with 3 important concepts that are the criteria for a good model:

1. a model of a believer must contain substructures or environments consisting of "relevant" items for a topic
2. those environments can correspond to other believers as well as to inanimate entities
3. the model of a believer must be able to nest environments to model the way the base believer can model the beliefs of other believers to any level required by the situation

4.2 The Importance of Environments

As previously stated, a good model should have some way of organizing beliefs into substructures or environments containing information relevant to a topic. Ballim [Ballim, 1992] supports the notion that *belief environments* can be used to organise beliefs and facilitate processes such as ascription, percolation, and relevance determination. A *belief environment* is a space permanently reserved for storing beliefs about X. For example, a belief environment about Tom would contain all the beliefs that Tom is believed to have as well as any other beliefs that are deamed relevant to him such as beliefs about Tom. A crucial advantage of the use of an enviroment is that it limits the reasoning space. For example, when comparing John and Marie's beliefs about dogs, it is only necessary to compare two limited environments. Without environments, it would be necessary to search all beliefs in the belief base in order to find the relevant beliefs to compare.

Environments are also very useful in maintaining consistency. With environments, it is not necessary or even desired to have a consistent belief base. Rather, it is necessary and desired to have consistent environments. This way, John's beliefs about dogs can be different, even contrary to Marie's beliefs about dogs.

Ballim has determined the need for at least three main types of environments that can provide an indexing scheme for beliefs. The first type of environment is the ***viewpoint*** environment which represents a particular agent's point of view. The second type is the ***topic*** environment which contains all the beliefs that are relevant to a given subject. In his thesis, Ballim states that "Topic environments are motivated by concerns of limiting reasoning. In short, it is envisaged that reasoning takes place 'within' a topic environment, as if it were the environment of a procedure in a [procedural -CG] programming language" [Ballim, 1992, p.6] or that of an object in an object-oriented programming language. The third main type of environment is the ***personae*** environment. It is used to represent the beliefs of typical classes of people. The only beliefs that would be maintained in a personae environment are those that differ from the system's beliefs.

4.3 The Importance of Nested Enviroments

In order for a belief model to be useful and handle real-world situations it must be able to model the beliefs one agent has about the beliefs of another agent. The model proposed by Taylor and Whitehill [Taylor and Whitehill, 1981], although it can handle mutual beliefs, provides no simple mechanism for representing beliefs that differ. For example, Jack believes that Jane believes his phone number to be 555-1212. Jane, however, believes that Jack has a phone number but does not know what it is. So, Jack's view of Jane's beliefs about his phone number would differ from Jane's view of her beliefs about his phone number.

Why is it important to be able to represent such differences? If a system could not represent the differences in beliefs such as this, there would be no way (or reason) to devise a plan to allow agent Jane to discover the phone number of agent Jack. For the purpose of this thesis, it is essential to have the ability to model beliefs held by the reader of news articles about all the sources cited in the articles. Ballim and Wilks [Ballim and Wilks, 1992] even claim that for machine belief to be considered belief at all, it must be able to model the beliefs of at least two distinct believers (the machine's beliefs and the beliefs of an interacting agent). This, of course, mandates the nesting of belief models.

The nesting of belief models is especially important for systems that must understand and/or participate in a dialogue. The following example, presented by Ballim [Ballim, 1992], provides good evidence of the need for nested models. Consider the following dialogue which must be understood by a system whose purpose is to allocate resources.

1. John: Sam is off sick today.
2. Sue: Then I can use the CAD terminal.
3. John: But you can't proceed without my layout.
4. Sue: I wasn't thinking about your layout, but about the one Mark did yesterday.

5. John: Oh, I didn't know he had finished it. [2]

After hearing (1), the system determines that the resources allocated to Sam are free for the day. The system can also interpret from (2) that Sue would like to have at least one of Sam's resources allocated to her (the CAD terminal). The third utterance (3) is a little more difficult to interpret. The system must have a model of John's beliefs and of John's beliefs about Sue's beliefs. John believes he has found an error in Sue's plan (that she can not continue without his layout). John believes that Sue needs his layout to continue and he also believes that she doesn't know this. Utterance (4) requires an even deeper nesting of models in order to be understood. The system must have a model of Sue's beliefs as well as Sue's beliefs about John's beliefs about her plans. For example, the system must model that Sue believes John believes she needs his layout, that Sue believes John believes that she does not know this and that she has an error in her plan, and that Sue believes that John is mistaken in believing that Sue wants to work on his layout. (5) requires that the system model John's beliefs about Mark and about Sue since this is John's explanation for why he thought Sue's plan was faulty.

So, for the purposes of understanding this short dialogue, the system must have a model of the beliefs of each participant in the dialogue (John, Sue, Mark, and Sam) as well as several nested models (John's beliefs about Sue's beliefs about her plans, Sue's beliefs about John's beliefs about Sue's plans, Sue's beliefs about Mark, and John's beliefs about Marks beliefs).

Clearly, a good belief model must allow the nesting of models so that a system can represent the beliefs of the system itself, of all the agents interacting with the system, and of the beliefs of the agents about the beliefs of the other agents. Furthermore, when modelling agents in an environment in which agents must cooperate or in a dialogue system, it is necessary to model the system's beliefs about the plans and goals of the agents interacting with it and of the agents' beliefs about the plans and goal of the other agents. In her work on plan inference, Pollack [Pollack, 1987] has shown the importance of models capable of representing the nested beliefs of agents. A method for modelling the plans and goals of agents within the ViewFinder framework has been proposed by Lee and Wilks [Lee and Wilks, 1996]. The models of agents in ViewFinder fall into the *Belief-Desire-Intention* (BDI) paradigm of modelling practical reasoning agents. (See [Georgeff *et al.*, 1999] for a discussion of the relative merits of BDI paradigm as compared with other modelling paradigms.)

4.4 Belief Diagrams in ViewFinder

ViewFinder is a framework for belief maintenance developed by Ballim [Ballim, 1992]. As part of this framework, Ballim has devised a representation for belief environments as shown in Figure 5. As described in Section 4.2, there are three main types of environments, namely: viewpoints, topics, and personae. Personae environments are not used for simple cases and will not be covered here.

A viewpoint environment is diagrammatically represented by a box with the agent's name in the center of the bottom line. Viewpoints can contain other viewpoints (i.e. John's view of Marie's viewpoint) or topics (i.e. John's view of dogs).

[2] Example taken from [Ballim, 1992, pp.4-5].

A topic environment can contain statements about the subject, in the form of propositions, or if the subject of the topic is an agent or class of agent, it can contain viewpoints.

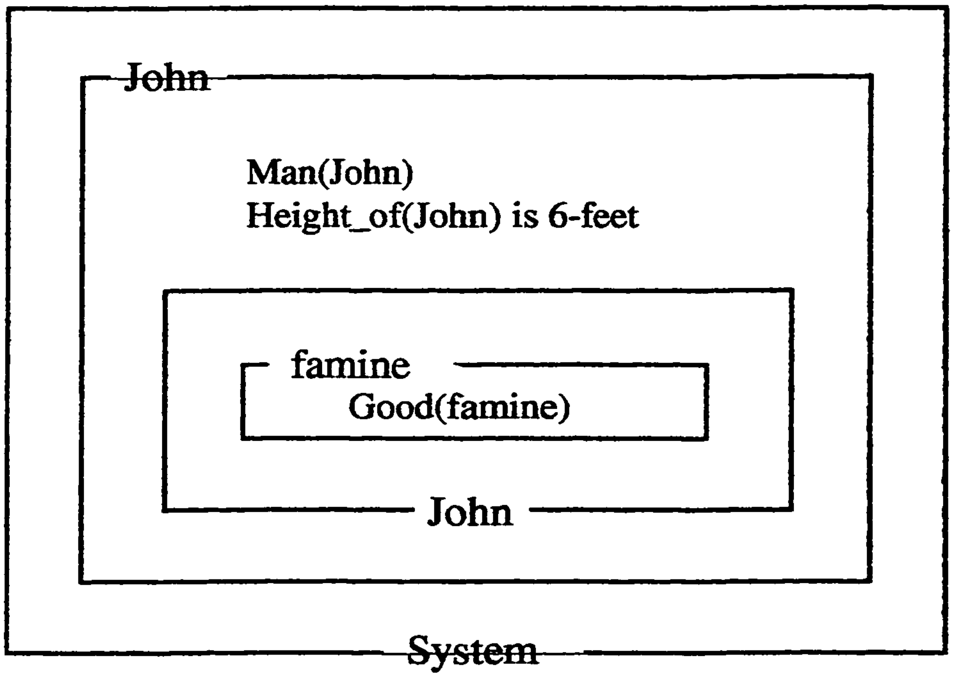

Figure 5: The System's beliefs about John's beliefs about famine

Figure 5 provides an example of how all of this can work together. A viewpoint is shown as a box with the name of the viewpoint holder in the centre of the bottom. This figure has two viewpoints: the system's and John's. A topic environment is shown as a box with the name of the topic on the upper-left-hand-side. This figure show's two topic environments: ***John*** and ***famine***. The ***John*** topic environment is contained in the system's viewpoint whereas the ***famine*** topic is contained in John's viewpoint. As depicted by this figure, topic environments in viewpoints other than the system's viewpoint, contain beliefs that conflict with the system's beliefs on the same topic. So, John's belief that famine is good, is shown here because it is different from or in conflict with, i.e. non-derivable, from beliefs in the system's topic environment (the system most likely believes that famines are bad).

Chapter 5

Belief Ascription in ViewFinder

In order to build a model of other agents, insight is needed into the goals, plans, intentions and beliefs of others. Much research has been done in the domain of agent modelling and user modelling. Most of the resulting theories and techniques have centered around determining the goals, plans, and intentions of other agents through direct evidence. By observing the actions of other agents over a period of time, it is sometimes possible to deduce what the agent is doing and why. Although it is possible to determine some of the beliefs of other agents through the observation and interpretation of utterances, called ***Belief Interpretation*** [Ballim, 1992], it is not possible to gather enough evidence to build an adequate belief model.

Ballim and Wilks contend that whether the agent be human or artificial, it is essential to have "a practical, heuristic theory for the ascription of belief to others" [Ballim and Wilks, 1992, p.5]. ***Belief Ascription*** is then defined as the "predictive attribution of whole groups of beliefs to agents based on various principles of commonality" [Ballim, 1992, p.57]. We can assume that others have the same set of common beliefs that we have. For instance, most people in developed countries believe that the Earth is round. It is, therefore, reasonable for John, an American, who believes that the Earth is round, to assume that Mary, also an American, has the same belief.

Through a combination of belief interpretation and belief ascription, it is possible to adequately model an agent's beliefs. In the following sections, three types of ascription will be described: attribution by perturbation, attribution based on stereotypical models, and the attribution of atypical beliefs. Another important concept introduced in [Wilks and Bien, 1983] is that of ***belief percolation*** which is presented as a side-effect of belief ascription. Section 5.4 will explain how the concept of belief percolation has evolved over time and how it relates to belief ascription.

5.1 Attribution by Perturbation

Attribution by perturbation is the action of ascribing or attributing a perturbed form of a belief from one agent to another. The simplest form of the attribution by perturbation algorithm is guided by a default ascription rule. This rule states that "X's view of Y can be assumed to be one's own view of Y, ***except where there is explicit evidence to the contrary***" [Wilks and Bien, 1983, p.103].

As shown in Figure 6, if the system believes the grass to be green, this belief would be ascribed to Tina because there is no explicit evidence that she believes differently. In terms of viewpoints, belief ascription is the attribution of beliefs from upper-level viewpoints to lower-level (nested) viewpoints.

According to the default ascription rule, beliefs can be attributed to another agent's view if there is no explicit evidence to the contrary. This means that when there is evidence that the other agent does believe something contrary, the ascription of the belief is blocked. This evidence can be direct negation of the belief in question or it can be an incompatibility based on the predicates (assuming beliefs are in predicate format) or functions (in the case of lambda functions). So, referring to Figure 6, if in the topic environment for grass in Tina's viewpoint, there had been a belief that Tina believed grass to be red, the ascription would have been blocked.

An advantage of belief ascription is that it helps to minimise redundancy of information because the same belief can be ascribed many times. It is not necessary to store the same information in several different environments. Furthermore, the dynamic aspect of belief ascription is important since it is impratical and wasteful to pregenerate all nested viewpoints. When it is necessary to have a fully instantiated viewpoint for a particular agent, one is generated on demand by the ascription mechanism. When the viewpoint is no longer needed, it is deconstructed. And as Ballim [Ballim, 1992] suggests, only the beliefs of other agents that contradict the system's beliefs should be kept. These should be stored long-term in the viewpoints of the particular agents. Consistency within a viewpoint must always be maintained to retain the usefulness of the enviroment (see Section 4.2).

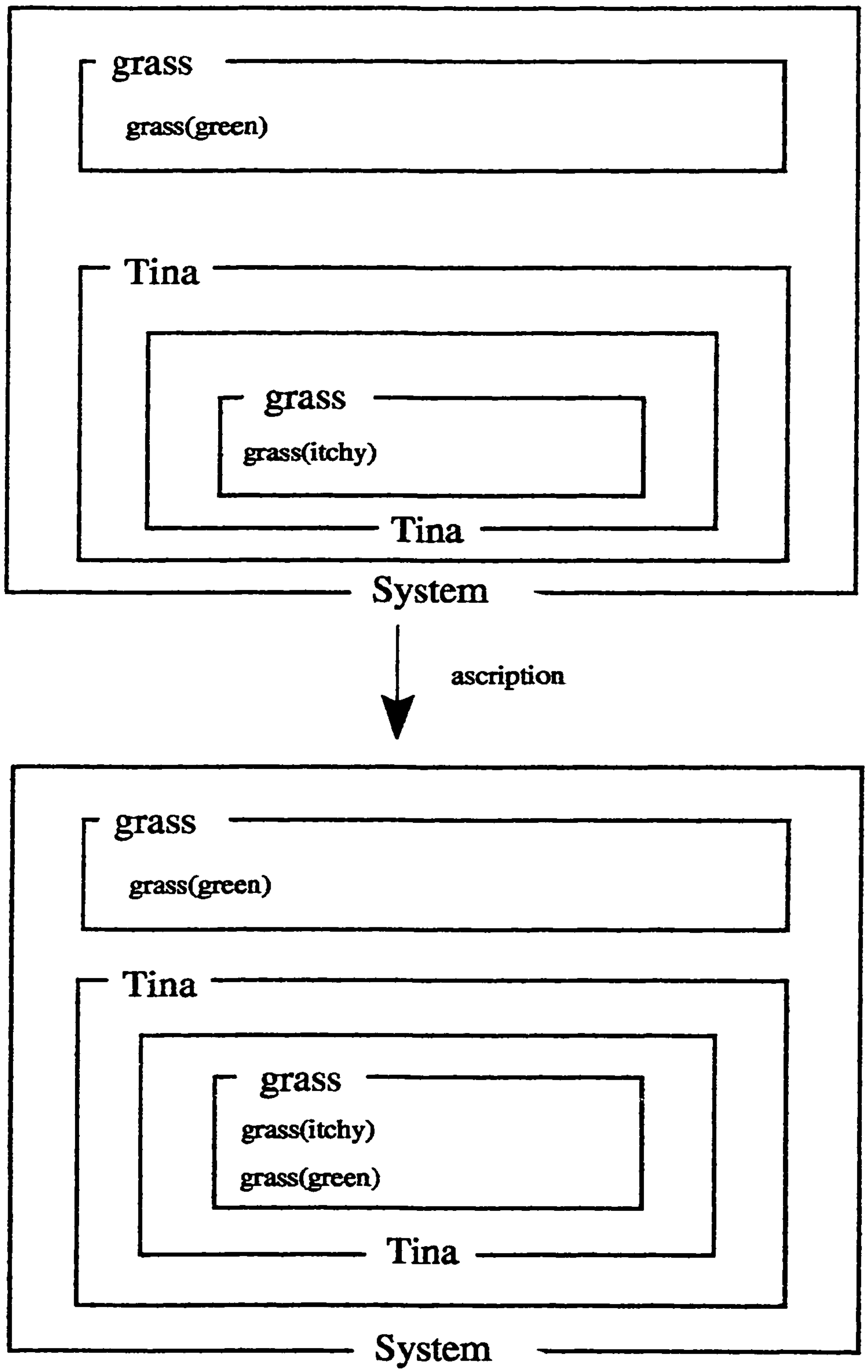

Figure 6: The Ascription of a Belief

5.2 Attribution by Stereotypes

A stereotype is a knowledge structure that contains information that is typically true of a group of agents [Ballim, 1992]. It is not a complete representation of the beliefs of *all* agents in a group. Instead, it is a characterisation of what is typical for that group. It typifies beliefs that are common to many members of a group but might not exactly fit every member of the group.

Stereotypical models, then, are models of *typical* beliefs that can apply to whole groups of people. These models are not dynamic. They must be pre-existent in a system. However, the application of a stereotypical model to an agent is dynamic. A sterotype has a set of clearly defined entrance criteria that need to be met by an agent for the agent to belong to that sterotype. The entrance criteria are often composed of three types:

1. *Necessary Preconditions* - a set of preconditions that must ALL be satisfied
2. *Sufficient Preconditions* - sets of preconditions such that ALL preconditions in one set must be satisfied
3. *Counter Conditions* - conditions or statements that if met, determine that a stereotype does *not* apply to an agent

Whether the agent meets the criteria or not can be established through direct questioning or interpretation of utterances. Once it is determined which stereotypes best fit an agent, the beliefs of those stereotypes can be ascribed to the agent [Ballim, 1992]. However, in order for a specific belief to be ascribed to an agent, it must not contradict any direct belief of the agent.

5.2.1 Conflicting Stereotypes

One or more stereotypes may be attributed to a particular agent. This can sometimes lead to conflicts between stereotypical beliefs. It is possible that it was erroneously determined that an agent belonged to a particular stereotype or that information from one stereotype conflicts with information from another. Several methods of dealing with such situations occur. [Rich, 1989] proposes the following methods:

- Numeric attributes indicating confidence can be attached to each belief.
- Forming a disjunction over the beliefs. However, as [Ballim, 1992] points out, two types of disjunctions can occur. We can ascribe a disjunction to the agent (the agent believes $P \vee Q$) or we can form a disjoined model of the agent (the agent believes P or the agent believes Q).
- Give priority to the sterotype that was first determined to apply to the agent.
- If one of the sterotypes was not previously involved in a clash, give it priority.

Ballim [Ballim, 1992] proposes an alternate method which consists of imposing a taxonomic hierarchy on the stereotypes. Such hierarchies are a well known technique in the field of Artificial Intelligence and the usual benefits also apply in this situation. A taxonomic hierarchy is a predetermined ordering, and thus, preference of the stereotypes. The possible benefits are as follows:

- provides ordering on stereotypes required for resolving conflicts
- reduces the number of stereotypes that need to be searched for an agent. Since the stereotypes lower down in the hierarchy are specialisations of those higher up in the hierarchy, all stereotypes above a triggered stereotype need not be considered.
- special binary search algorithms can be developed to search on the hierarchy.

However beneficial, the use of taxonomic hierarchies of stereotypes will not prevent all types of conflicts. As with most complex inheritance schemes, multiple inheritance is often necessary. And because sterotypes do not apply completely to every member of the group, ambiguities will occur. If too many such ambiguities occur, it might be necessary to introduce a more fine-grained inheritance reasoning system for stereotypes.

5.3 Attribution of Atypical Beliefs

Wilks and Ballim [Wilks and Ballim, 1987, Ballim, 1987] propose a special class of beliefs called ***atypical*** beliefs. These are beliefs that can be held by an agent but would not normally be held by all agents. This class of beliefs would include self knowledge, secrets, expertise and knowledge about uncommon domains. However, Ballim [Ballim, 1987] cautions that a belief can only be considered atypical with respect to a particular agent if the agent holding the belief also believes it to be atypical. For example, John may believe that Mary has an atypical belief. But, John can also believe that Mary believes the belief to be typical. So, John believes the belief to be atypical, i.e. not commonly held by other agents, whereas Mary believes it to be typical, i.e. commonly held by other agents. So, when ascribing beliefs to another agent through Mary's viewpoint, the atypical belief would be ascribed as though it were a typical belief.

However, when a belief is believed to be atypical, it is ascribed differently from typical beliefs. In order to ascribe atypical beliefs, a rule opposite to the default rule must be applied. That is, the rule for atypical beliefs should be "*not* to ascribe unless one has explicit evidence to justify ascribing the belief" [Wilks and Ballim, 1987, p.120]. For example, Mark is a doctor and he believes that a thyro-glossal duct cyst is a congenital defect. If we were generating Mark's beliefs about Mary who is an engineer, the ascription algorithm would not attribute this belief to Mary because there is no explicit evidence that she would have this type of atypical knowledge. However, if Mary were also a doctor, that would be sufficient evidence to attribute the belief to her.

Wilks [Wilks, 1986] observed that stereotypes could, in fact, be viewed as a type of atypical belief. An agent to which an atypical belief or a stereotypical belief applies would be one with the ***competency*** to hold that belief. Wilks also proposed the use of ***lambda expressions*** with restrictions on the capable evaluators of the expression as ways to denotate the necessary competency.

(4) $\exists\_X \{X = (\lambda.L(immed\_type\_of\,L)thyro\_glossal\_duct\_cyst) < MDs >\}$

Definition (4) states that there is an evaluation of the lambda expression such that a thyro-glossal-duct-cyst is an immediate type of L. The only people known to be able to evaluate the lambda expression, who know that a thryro-glossal-duct-cyst is an immediate type of ***congenital defect*** are MDs. This seems simple enough but there are cases where it is not so straight-forward. There are various aspects of competency that can be expressed as a taxonomy. For instance, we all know that people are capable of considering beliefs involving such atypical beliefs without having the competence to evaluate the belief. In addition, it is possible that two MDs could evaluate the lambda expression to different values. Figure 7[1] borrowed from [Ballim, 1992] shows a taxonomy of the types of competencies that an agent can have with respect to lambda expressions.

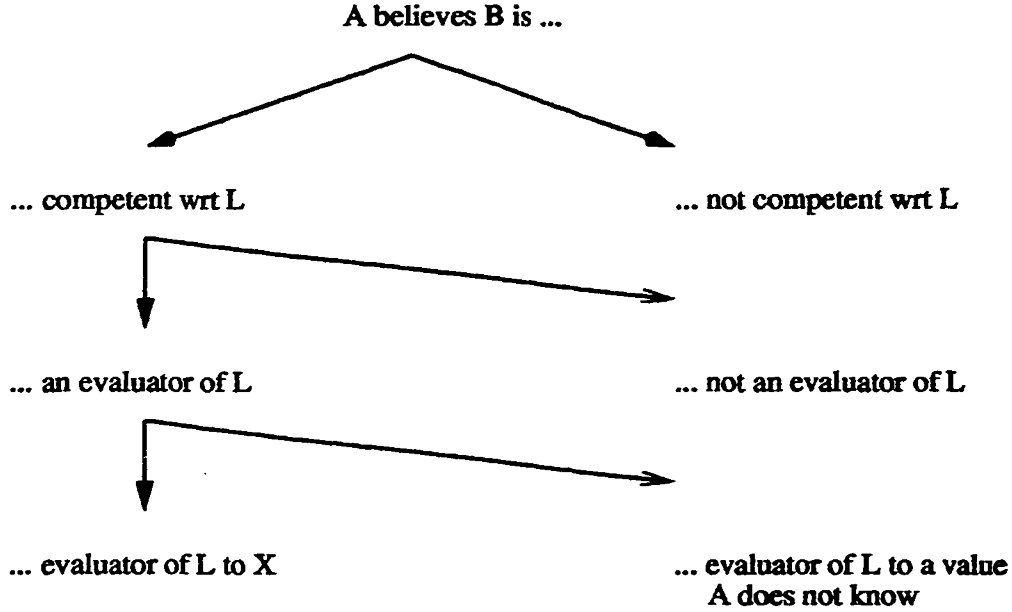

Figure 7: Taxonomy of Competencies for Lambda Expressions

The following example taken from [Ballim, 1992] illustrates the consequences of this taxonomy clearly. Consider a belief about John's phone number P that contains a lambda expression L. It would be read as a belief that John's phone number is some value and expressed as in Equation (5).

(5) $\exists .X\ \{X = (\lambda .L(phone\_number\_of L) John)\}$

As shown in Figure 7, the belief P would not be attributed in any form to someone who has never heard of John, because they don't have competency with respect to L. However, people who do know John, have the competency to hold beliefs about John such as his phone number. We would ascribe the belief P to them because they have the competency to have the lambda expression L and can be evaluators of L to a value X.

There are also situations in which people have the competency to hold the lambda expression L without being able to evaluate it. For example, Peter can believe that Rachel has beliefs about John's phone number even though he knows that she doesn't know what it is. In this case, Rachel has the competency to hold the lambda expression L even though she is not an evaluator of L. It is

[1]Figure 8.1 in [Ballim, 1992]

also possible to have a situation where Peter does not know what John's phone number is but he believes that Rachel believes that she knows John's phone number. Here, Rachel has the competence to hold the lambda expression L and is an evaluator of L to a value that Peter doesn't know.

As has been shown by the previous example, "competency is not simply a case of deciding whether or not to ascribe a belief, it may involve ascribing some transformed version of a belief" [Ballim and Wilks, 1992, p.171].

The use of the lambda expression can then be taken one step further by associating with each lambda expression a complex evaluation relation expressing the competency of the agents in evaluating the expression. In addition, it specifies how the agents will evaluate the expression. This is called a ***lambda formula*** [Ballim, 1992]. For example, the lambda expression in (Equation (5) becomes the lambda formula in Equation (6).

$$(6) \quad \begin{array}{l} (\lambda.L(phone\_number\_of\,L)\,John) \\ comp\left(\begin{array}{c}[(\text{system}),5554992], \\ [(\text{Mary}),9995834]\end{array}\right)\end{array}$$

Here, the complex relation *comp*(X, Y) states that there are two ***competing*** evaluation relations X and Y for the evaluation of John's phone number. The system believes the number to be 5554992 whereas Mary believes it to be 9995834. Ballim [Ballim, 1992] also proposes two other evaluation relations. One is *spec*(X, Y). It indicates that the agents involved in the evaluation relation X believe their evaluations to be more specific than the evaluations derived in evaluation relation Y. At the same time, it indicates that the agents involved in evaluation relation Y do not believe this to be the case. Another complex relation is *specK*(X, Y). This is a variation of *spec*(X, Y) in which the agents in Y believe that the agents in X have a more specific evaluation even though they might not know what that evaluation is.

5.4 Belief Percolation in ViewFinder

This section will explain the concept of belief percolation and how it is important in representing how a system can acquire beliefs and assume these as beliefs of its own. An agent that can acquire new beliefs exhibits the most important characteristic of so-called ***adaptive autonomous agents*** [Maes, 1995], that of ***learning*** from its own experience or that of other agents. It will also show how the meaning of this concept has changed since its introduction in 1983 by Wilks and Bien.

5.4.1 Pseudo-Text Structures and Percolation

In an early formulation of the ascription mechanism [Wilks and Bien, 1983], pseudo-text (PT) structures were used instead of environments. A PT was very similar to an environment in that it contained the beliefs of an agent or the beliefs of an agent about another agent. Initially, the system would have only top-level or "outer" PTs. For example, the system might have PTs to represent the system's beliefs about the user or the system's beliefs about Frank. These would be represented as in Figure 8. In order to create the system's view of the user's view of Frank, the system would first

{FRANK} *SYSTEM*   *{USER}* *SYSTEM*

Figure 8: Top-Level PTs for the User and Frank

construct or locate the system's view of Frank and then the system's view of the user. Then, the system would "push" the systems view of Frank into the system's view of the user. The resulting push-down would be represented as in Figure 9.

$$\left\{ \begin{array}{l} \{FRANK\} \\ USER \end{array} \right\}_{SYSTEM}$$

Figure 9: The System's view of the User's view of Frank

In the terminology of Wilks and Bien [Wilks and Bien, 1983], if an outer proposition migrates to an inner PT, the proposition has *percolated*. An important consequence of this *percolation* is that once the generated views are no longer needed, they are decomposed. This means that in some cases, beliefs that became inner beliefs can now become outer beliefs in a different PT. Percolation can then be defined as "a method in which beliefs propagate about a belief system in a way not necessarily intended by any believer or participant, but which follows as a side effect of our principal algorithm" [Wilks and Bien, 1983, p.96]. The following example, borrowed from [Wilks and Bien, 1983] illustrates how beliefs can *percolate* about a belief system.

Figure 10 shows the two PT's from Figure 8 populated with some beliefs. The "*" is used as a reference to the owner of the PT. For instance, in the PT for Frank, "user likes *" means that the system believes that Frank believes that the user likes Frank.

$$\left\{ \begin{array}{l} FRANK \\ \text{user likes *} \\ \text{* dislikes user} \end{array} \right\}_{SYSTEM} \qquad \left\{ \begin{array}{l} USER \\ \text{*dislikes Frank} \\ \text{Frank dislikes *} \\ \text{Frank dislikes system} \end{array} \right\}_{SYSTEM}$$

Figure 10: Populated PTs for the User and Frank

If the system wanted to create the PT for the user's view of Frank's view, the system would first locate or generate the two PTs in Figure 10. Then the system would need to push the PT for Frank into the PT for the user. Only the belief "* dislikes system" would be pushed into the resulting PT for the user's view of Frank's view because the other two beliefs in the PT for the user are blocked by the beliefs in the PT for Frank. Figure 11 shows the result of the push-down operation.

An attempt is made to push-down each belief in the system's view of the user into the system's view of the user's view of Frank. However, the belief "User dislikes Frank"is blocked by Frank's

```
{ USER
  { FRANK
    * dislikes system
    user likes *
    * dislikes user
  } SYSTEM
  * dislikes Frank
  Frank dislikes system
  Frank likes *
} SYSTEM
```

Figure 11: The System's view of the User's view of Frank's view after Push-Down

belief that "user likes Frank". The belief that "Frank dislikes user" is already held by Frank and as such does not need to be pushed down. However, the user's belief that "Frank dislikes system" is not contradicted by any belief in Frank's environment and is pushed-down into Frank's view slightly transformed into "* dislikes system". Later, when this nesting is no longer needed, it is deconstructed into the system's view of the user and the system's view of Frank as in Figure 10. The only difference is that now the belief that "Frank dislikes system" will appear in the top-level PT for the system's view of Frank. The belief will have percolated from an inner PT to an outer PT.

The most important aspect of percolation as described here is that beliefs become held without direct evidence for being believed. In this example, the system has no reason for believing that Frank dislikes it. In addition, once the nesting is deconstructed, the system no longer even has the source of the belief and so the belief is believed to the same extent as a belief for which the system has evidence.

In later writings [Wilks and Ballim, 1987], [Ballim, 1992], [Ballim and Wilks, 1992], this process was divided into two concepts. The mechanism allowing beliefs to move deeper into nested environments is described as belief ascription (see Chapter 5) which is guided by the default ascription rule. This rule states that "X's view of Y can be assumed to be one's own view of Y, *except where there is explicit evidence to the contrary*" [Wilks and Bien, 1983, p.103]. The side-effect of belief ascription, that of beliefs propagating from inner environments to outer environments, is described as belief percolation. Wilks and Ballim claim that percolation "corresponds to real ways in which unsupported beliefs are propagated among believers" [Ballim and Wilks, 1992, p.243] and that it is supported in psychology as the well-attested "sleeper effect" (Gruder et al., 1978) in which subjects come to hold beliefs for which they have no evidence as side-effects of other processes" [Ballim and Wilks, 1992, p.243].

Chapter 6

Belief Percolation as a Learning Process

As was shown in Section 5.4, Ballim, Wilks, and Bien [Wilks and Bien, 1983, Wilks and Ballim, 1987, Ballim, 1992, Ballim and Wilks, 1992] consider belief percolation simply as a side-effect of the belief ascription mechanism, almost an after-thought. In fact, the meaning of this concept has evolved since its introduction by Wilks and Bien. As such, this appears to be an area of study in need of further investigation and development. I believe that belief percolation can provide an adequate expression of the learning or acquisition of beliefs.

This chapter shows that the notion of belief percolation can be developed into a mechanism that will allow user models to represent what is learned by an agent through the reading of newspaper articles. This new mechanism will consist of three main elements: a ***default percolation rule***; a ***belief reliability heuristic***; and a ***source list***. In addition, Ballim's belief diagrams (see Section 4.4) will be adapted to more intuitively represent how beliefs are acquired according to this new mechanism.

6.1 Motivation

This thesis is concerned with modelling how a reader can acquire beliefs from the reading of news articles. Thus, the views to be modelled are those of agents such as the system, the reader, the reporter, and any sources mentioned in a given newspaper article.

Given the goal of this thesis, modelling the acquisition of beliefs, having belief percolation be a simple side-effect of belief ascription will not suffice. Rather, belief percolation needs to be modified and developed into a full-fledged mechanism. This mechanism will be, in essence, the reverse of Ballim's belief ascription and will rely heavily upon the notions of competence and atypical beliefs. As was explained in Section 5.3, the rule for the ascription of atypical beliefs is that atypical beliefs in ViewFinder ***should not be ascribed unless there is explicit evidence to justify doing so.*** Atypical beliefs are represented as lambda expressions or lambda formulas that specify particular agents or classes of agents who are capable evaluators of the expression i.e. who have the competence in the subject of the belief to evaluate it.

This is very similar to the case of news articles. The sources in news articles have (or should have) expertise, authority or involvement in the subject of the belief which makes them good sources. If their beliefs about the subject are typical then there is no news to be reported. What makes a source a good source is the atypical nature of his beliefs. That being said, these beliefs could not, by any mechanism described thus far, be attributed to the reader who most likely is not an evaluator of the atypical beliefs held by the sources. Belief percolation needs to be adapted to this special situation to allow agents even though they may not be competent in the subject matter to acquire atypical beliefs about the subject.

Belief ascription can not be used to represent the acquisition of beliefs by readers of news articles because in belief ascription, beliefs from outer environments are pushed into inner environments. The default rule for belief ascription is "X's view of Y can be assumed to be one's own view of Y, *except where there is explicit evidence to the contrary*" [Wilks and Bien, 1983, p.103]. This rule must be considered in terms of nested beliefs where the reader can assume that the reporter's view of the topic is the same as his own and likewise, the reporter can assume that the source's view of the topic is the same as his own unless there is explicit evidence that the other has a contradicting belief. Belief ascription is concerned with propagating those beliefs which are commonly held by the owner of the outer viewpoint to an inner viewpoint to create a view of a viewpoint. When the reader creates his view of the reporter's viewpoint, the reader's beliefs will be ascribed to the reporter, not the other way around. In the case of belief acquisition, the concern is not commonly held beliefs but rather atypical beliefs. Furthermore, the focus is not on creating the reader's view of the reporter's view of the topic (although initially this must be done). Rather, the focus is on propagating the source's atypical beliefs to the reader.

Ballim and Wilks deny that the notion of default belief ascription creates a situation in which learning is not possible since "ascription models the beliefs of others, and can not in any straightforward way constrain basic belief acquisition or rejection of it by the system itself" [Ballim and Wilks, 1992, p.181]. Rather, they claim that the belief ascription, deconstruction of the viewpoint nesting and belief percolation cycle, indeed, represents learning [Ballim and Wilks, 1992]. In the ViewFinder framework, the system viewpoint contains many topic environments which constitute the system's knowledge-base. The system ascribes the beliefs in its topic environments to the nested models. ViewFinder was initially intended as a framework for a dialogue system. As such, inner models can contain beliefs that are not part of the system's knowledge-base. The system becomes aware of these beliefs through its interaction with other agents. During the deconstruction of a nesting of viewpoints, inner environments become outer environments in the system's viewpoint causing inner beliefs to percolate to the outer environments. If the system has confidence in the source of a new belief, the belief will remain in the standard copy of that environment when it is reestablish in the system's viewpoint. In this way, the system will have "learned" from the other agent.

Belief percolation is a very important mechanism for the modelling of readers of news articles. However, it can not be used as described by Ballim and Wilks for several reasons. First, belief percolation in ViewFinder is not fully developed yet. Many questions still remain as to exactly which types of beliefs can be percolated and when percolation is appropriate. Secondly, ViewFinder was created for an entirely different kind of application, i.e. a dialogue system. The system creates models of the other agents in order to be able to dialogue with them. The system is then able to "learn" through dialoguing with the other agents. When modelling readers of news articles, there is no dialogue with other agents. Rather, the nested viewpoints are created based on information in a news article (although the viewpoints can receive additional beliefs from belief ascription). Thirdly, it is the reader model that is of most concern. It is the reader model that must "learn" from the sources in the news article. Even assuming that the system could learn from the sources using the ascription-deconstruction-percolation cycle as described by Ballim and Wilks, the system could not in a later nesting ascribe the newly learned beliefs to the reader since the beliefs would most likely be atypical beliefs and the reader would probably not have the competence to be an evaluator of the beliefs. To adequately model how readers of news articles acquire beliefs from sources in the articles, belief percolation needs to be made the reverse of belief ascription, percolating atypical beliefs through the viewpoint nestings from the inner environments to the outer environments.

As described earlier, when ascribing atypical beliefs, beliefs are often ascribed in a modified or transformed fashion based on the competency of the receiving believer. The transformation is detailed in the lambda formula. To some extent, the same is true of percolation. Given a reader, ignorant of the workings of the stock market, who has been reading an article about the finer points of the stock market the source of which is a Wall Street wizard, it would be reasonable to assume that the reader would acquire some beliefs about the stock market although his understanding of those beliefs would not be as profound as that of the source. Since the source's beliefs are atypical, although they are not represented by lambda formulas, it is reasonable to assume that the beliefs would have been somewhat transformed during percolation by the competency or lack of competency of the reporter and the reader himself. The reporter may misunderstand what the source is explaining to him, or if the reporter is knowledgeable in the subject matter, he might embellish the source's comments adding his own flavour to them. In addition, the reader might do the same. It is a common experience that two people read the same text and later, when discussing it, realise that they have a very different understanding of what they have read.

Furthermore, percolation as described by Wilks, Bien and Ballim [Wilks and Bien, 1983, Ballim, 1992] only allows for the source of a belief to be traceable while the nesting of environments still exists (by following the nestings, the source can be identified). Once the nesting of environments is decomposed, beliefs in the inner environments become outer beliefs, i.e. beliefs of the system about someone or something and the system no longer knows how it acquired the belief.

This is undesirable for modelling the reading of news articles for several reasons. First, beliefs acquired through percolation are by their very nature tenuous. They are acquired without the agent having personal knowledge of the subject and as such are more likely to be proven false. The fact that there is no way to know where the belief came from makes belief maintenance of this type of belief extremely difficult if not impossible. If a belief needs to be revised, the belief revision needs to be propagated through the affected environments. To do this, it must know through which viewpoints the belief was percolated. Secondly, it is not very representative of how humans acquire beliefs. Generally, we can for some time remember the source of a belief especially in cases where we did not see proof of the belief. People will often say, "I read in the Gazette that ..." or "A friend told me that ...". In this way, people are saying, "I'm telling you this but I don't yet know if it's true." The percolation mechanism proposed here will allow for the trace of a percolated belief to be maintained until the reader has sufficient cause for accepting the belief as one of her own.

The mechanism proposed here will provide for (1) a distinction between newly acquired, uncertain or *potential* beliefs and accepted beliefs, (2) a method of keeping a trace of the sources of the belief, and (3) a belief heuristic for determining when a newly acquired, *potential* belief becomes a *held* belief.

6.2 Modified Belief Diagrams

A reader of news articles will be presented with many "beliefs" in an article. It would not be valid to assume that the reader will accept all of these "beliefs" as true. Rather, in most cases, the reader will wait to get more corroborating information about the belief. However, Ballim's belief diagrams do not allow for any distinctions among the beliefs of an agent. In order to model the beliefs of a reader of news articles, it is necessary to be able to distinguish between those beliefs that the reader *accepts as true* and those about which the reader is waiting to get more corroborating information. This section will show how Ballim's belief diagrams can be modified to represent this unique situation.

As was explained in Section 4.4, Ballim's viewpoint environments can contain either other viewpoints or topic environments. Topic environments, in turn, can contain statements about the topic or other viewpoint environments. Beliefs *about* the owner of a viewpoint are placed in the topic environment for that owner and beliefs *held by* the owner are placed in the viewpoint environment. In Figure 5, the belief man(John) is a belief about John held by the system. It is placed in the topic environment for John in the system's viewpoint. The belief good(famine) is a belief held by John about famine. It appears in the topic environment for famine in John's viewpoint.

In Ballim's diagrams, there is no distinction between beliefs that John believes firmly (i.e. has evidence for) and those which he has only come to believe through such mechanisms as percolation (i.e. has no direct or personal evidence for). However, when modelling the belief acquisition process, it is important to be able to distinguish beliefs that are firmly held from those that are merely potential. Readers, especially of newspaper articles, are presented with hundreds of potential beliefs everyday, not all of which become held beliefs. Readers generally filter much of what they read based on certain factors such as their own set of beliefs, their competency relative to the subject, and their confidence in the reporter and the source. For instance, if a democrat reader is reading about how a democrat senator accepted bribes from members of a foreign government, she will be much more likely to acquire this belief if the reporter and/or the source are also democrats. One reason for this is that the reliability of the statement is increased when the source is conceding to something that is detrimental to himself. In other instances, the reliability of the belief increases with the number of times we receive the same information from different sources.

For the purposes of this thesis, it is necessary to distinguish the two types of beliefs. This is done by modifying Ballim's belief diagrams such that beliefs in the topic environments in the viewpoints are separated by a line into held beliefs and potential beliefs. Here, we are only concerned with this distinction for the reader. As such, only the topic environments in the ***reader's*** viewpoint will be separated into potential and held beliefs.

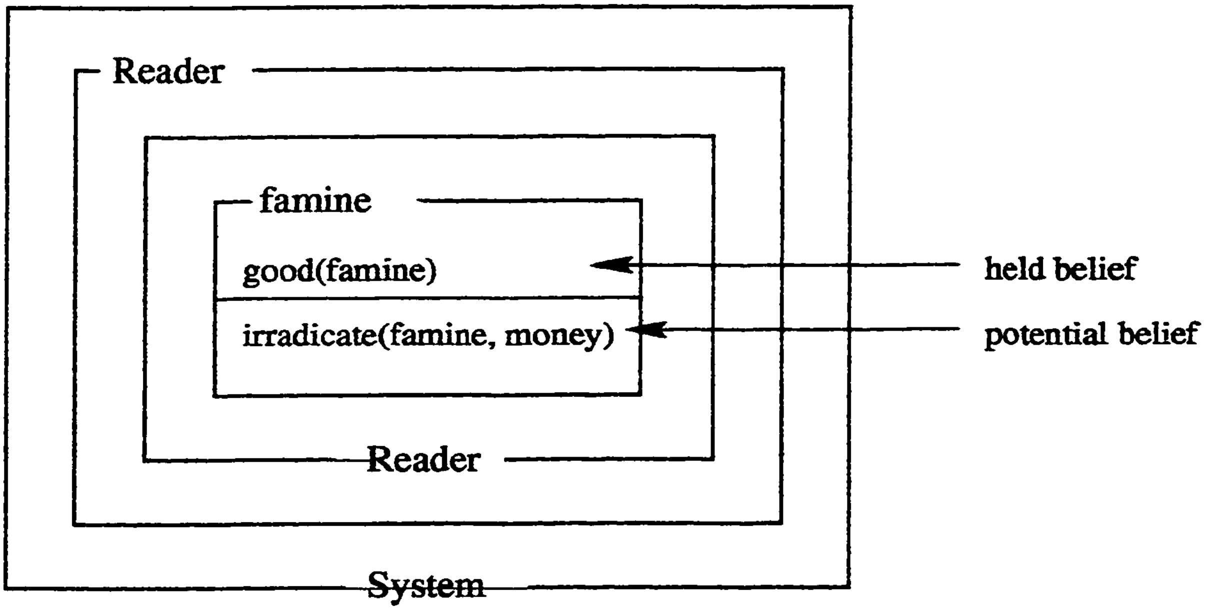

Figure 12: Modified Belief Diagram

Figure 12 depicts the system's view of the reader's viewpoint about famine. The reader has the ***potential*** belief that famine can be irradicated by money. The reader does not yet accept this belief as true. Rather, he is waiting to get some corroborating information about the belief, i.e. read it from some different sources. The reader also has the ***held*** belief that famine is good. This is a belief that the reader accepts as true. He could have acquired it in a variety of ways. It could be a belief that he acquired through belief ascription. Or, it is a belief that was acquired through belief percolation and was later corroborated. Obviously, it is a belief that could still be false. If the reader is presented with new information about the goodness or badness of famine, this held belief can be revised.

Figure 12 shows the System viewpoint containing a Reader topic environment. The system, however, has no beliefs ***about*** the reader and so, it is not necessary to nest the reader's viewpoint within a Reader topic environment. It is shown here for illustrative purposes. The surrounding topic environments will not be shown in all other diagrams unless expressly needed.

6.3 Default Percolation Rule

Belief percolation as redefined by this thesis relates to Ballim's treatment of atypical beliefs [Ballim, 1992]. As a reminder, an atypical belief is a belief that can be held by an agent but would not normally be held by all agents such as a belief involving self knowledge, secrets, expertise or knowledge about uncommon domains. In Section 5.3, it is stated that the rule for the ascription of atypical beliefs is to only ascribe an atypical belief if there is explicit evidence to justify the ascription. The default percolation rule proposed here incorporates this rule for atypical beliefs since only beliefs from reliable sources will eventually become held beliefs of the reader. However, in the case of percolation, the application of the rule is less strict than for the ascription of atypical beliefs since beliefs are initially percolated into the reader's environment as *potential* beliefs only. Potential beliefs do not participate in any ascription activities that may occur as the result of other events in the system.

Therefore, the default percolation rule is to ***percolate beliefs as potential beliefs when there is evidence to justify doing so.*** The justification in the case of a system modelling readers of news articles is that the reader is reading articles on topics or events with the intent of gaining information. The sources of the information are supposed to be competent with respect to the information. In order to safeguard the reader's viewpoint against unreliable information or information contradictory to beliefs already held by the reader, beliefs are percolated as potential beliefs.

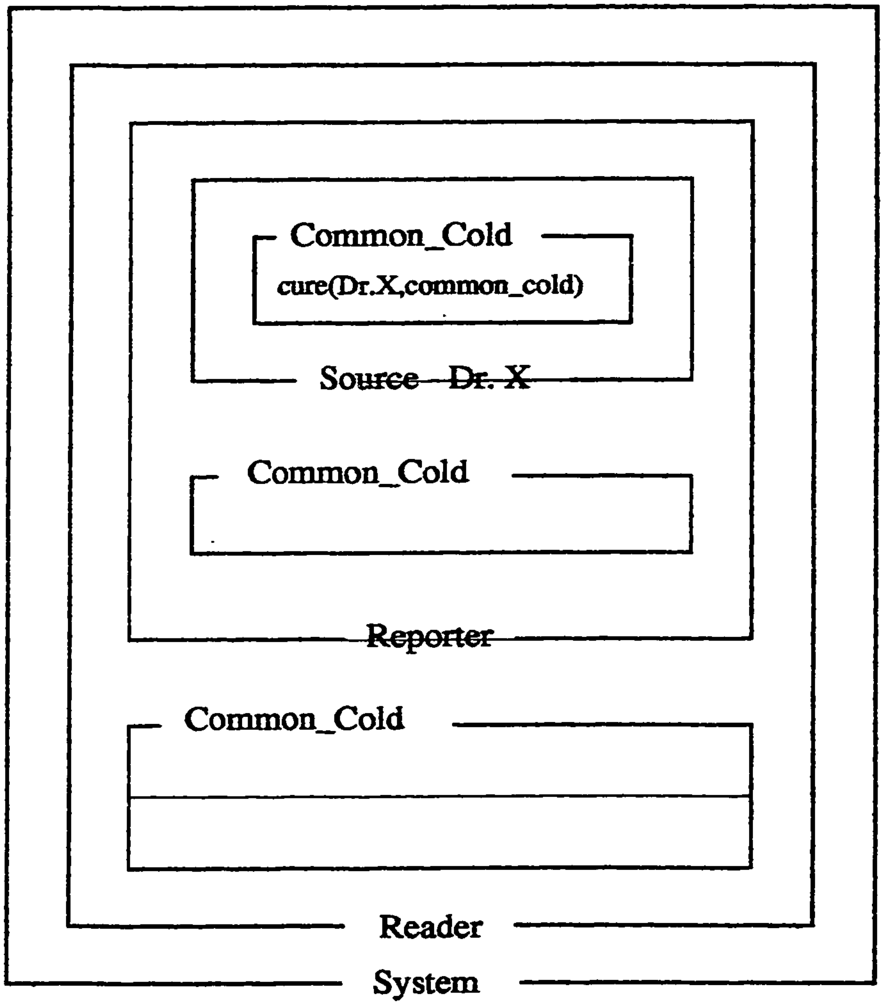

Figure 13: Example of Percolation - Source has Belief

Figures 13, 14, and 15 collectively demonstrate how the default belief percolation rule can be applied to simulate the acquisition of beliefs by the reader of news articles. The example used for these figures is an article about a certain researcher, Dr. X, who believes that he has found the cure to the common cold. Viewpoint environments for the main agents in this example, Dr. X, the reporter, and the reader, are shown. Also, for each agent, there is a topic environment for Common_Cold.

Figure 13 depicts the initial state of the viewpoints. Only the source, Dr. X, has any belief concerning the common cold. The Common_Cold topic environment in the source's viewpoint contains the belief cure(Dr. X, common_cold). The Common_Cold topic environments for the reporter and the reader are empty denoting that neither has any beliefs concerning the common cold. The source's topic environment for Common_Cold is not separated into held and potential beliefs since this thesis is only concerned with modelling the reader's acquisition of beliefs. Furthermore, although it would be reasonable to assume that Dr. X believes this belief as a held belief, it simplifies things to make no judgement about the status of this belief for the source.

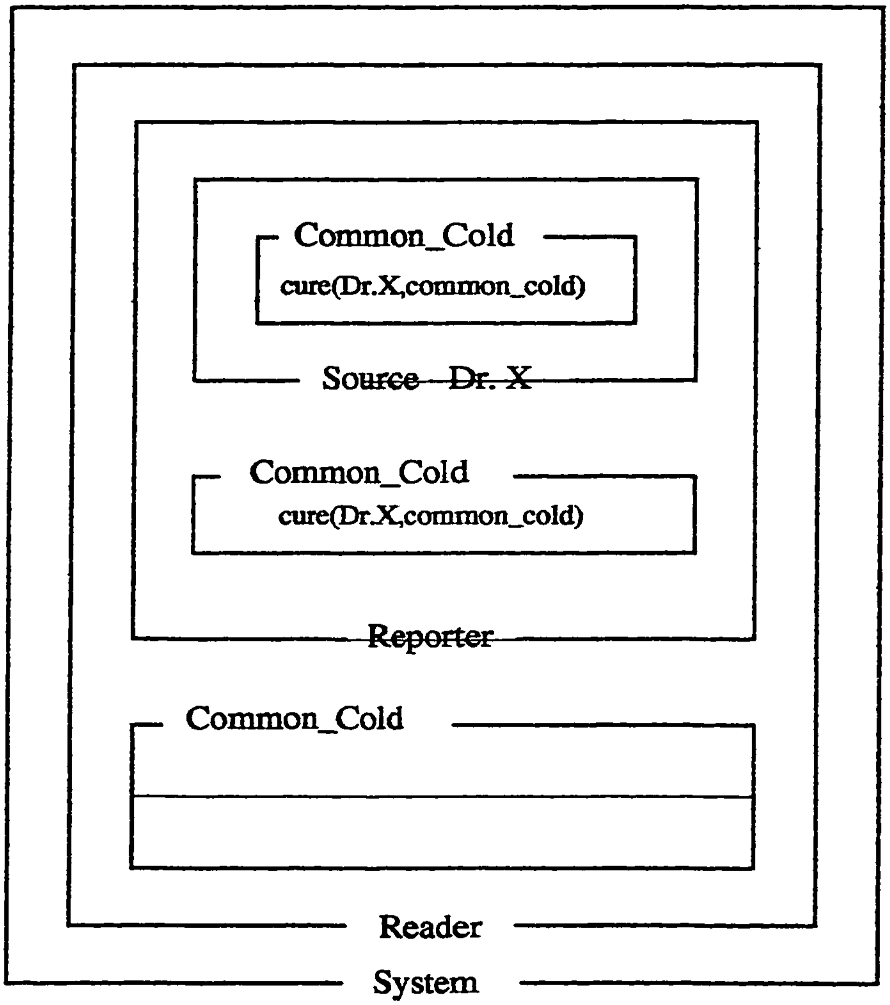

Figure 14: Example of Percolation - Reporter has Belief

Figure 14 shows the state of the viewpoints once the belief that *Dr. X has the cure for the common cold* has been percolated into the reporter's viewpoint. The reporter's topic environment for Common_Cold contains the belief cure(Dr.X, common_cold). The belief has been percolated without transformation in order to simplify the example. As previously stated, the reporter's topic environment for Common_Cold is not divided into held and potential beliefs since the reporter's viewpoint is not the main concern of this thesis. Furthermore, there is no evidence indicating whether this is a held belief or a potential belief for the reporter.

Figure 15 shows the state of the viewpoints that concerns this thesis the most. The belief has been percolated into the reader's viewpoint. Again, the belief has been percolated without transformation in order to simplify the example. The topic environments of all three agents, the source, the reporter, and the reader, contain the belief cure(Dr. X, common_cold). However, in the case of the reader, the belief is percolated into the lower-half of the Common_Cold topic environment indicating that it has been percolated into the reader's viewpoint as a *potential* belief only.

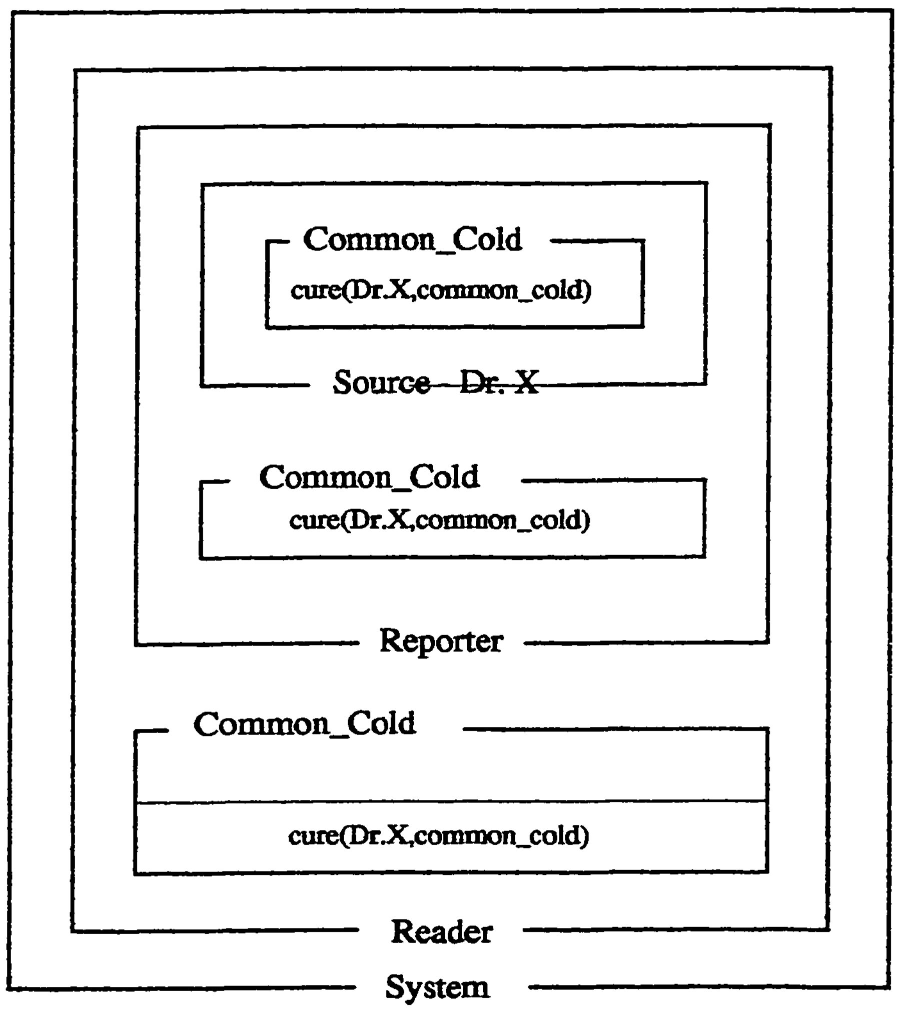

Figure 15: Example of Percolation - Reader has Belief

As a second step to the percolation process, potential beliefs are evaluated using a belief heuristic which determines the reliability of a belief based on the reporter's confidence in the source, the reporter's confidence in the reported information, the reader's confidence in the source and the reader's confidence in the reporter. If the belief is deemed reliable, it will become a held belief as explained in Section 6.4.6. In the case where the reader's viewpoint contains a contradictory belief, the new belief will override the old belief and the belief maintenance mechanism will be invoked to remove any beliefs that may be dependent on the old belief.

I contend that this process is consistent with how people acquire new beliefs and discard old ones. A person may read some information and not initially believe it. After several exposures to the same information from various sources, the belief becomes more reliable and at some point becomes fully accepted by the reader. This also appears to be true in situations where the new belief reverses some previously held belief. This is further exemplified in Section 6.5.4.

6.4 Belief Reliability

Belief reliability expresses the reader's confidence in the credibility of a belief. The level of reliability is calculated by a ***belief heuristic***. The belief heuristic determines the reliability of a potential belief based on the following factors: (1) the reader's confidence in the source, (2) the reader's confidence in the reporter, (3) the reporter's confidence in the source, (4) the reporter's confidence in the credibility of the reported information, and (5) the reader's previous beliefs about the topic of the potential belief. The first four factors will, henceforth, be called the ***confidence criteria*** and the fifth factor will be called the ***opposing belief criterion***.

The first two confidence criteria can only be evaluated within a belief environment since they depend on the current state of the reader's beliefs about the source and the reader's beliefs about the reporter. Likewise, the opposing belief criterion can only be evaluated within a belief environment since it depends on the current state of the reader's beliefs about the topic of the belief. The third and fourth criteria, however, must be evaluated during a phase of evidential analysis which needs to occur prior to converting the text into logical proposition-like structures. This is because, as was discussed in Section 3.1.1, only the ***primary information***, the information needed to push the story along usually provided by an expert or witness, is maintained during the conversion. However, it is the ***circumstantial information*** that provides the details for evaluation. As mentioned in Section 3.1.1, the circumstantial information gives details about the who, when, and how surrounding the primary information. The following subsections will explore each of the criteria in more detail.

6.4.1 Reader's Confidence in the Source

Many researchers in the field of belief maintenance systems order beliefs in the system according to their level of credibility. Johnson and Shapiro [Johnson and Shapiro, 2000] have also recognised the need to obtain a credibility assessment for the source of a belief. However, the credibility assessment is obtained directly from the user. The user needs to input information into the system declaring source credibility orders of the form Greater(Source1,Source2) which means that Source1 is more credible than Source2. Consequently, any propositions (beliefs) reported by Source1 are more credible than those by Source2.

Although ordering information could be used by the system proposed in this thesis to determine which of two opposing beliefs in one article to believe, the ordering method has several disadvantages. First, it relies on input from the user. This method disturbs the user and brings up a dilemma: should the user provide a one-time ordering? If so, then the ordering may become incorrect after some time as the user's beliefs about the sources may change. If not then, the user will need to provide ordering information every time a new source is introduced or or his beliefs about a source change. This may occasion substantial re-ordering of all the sources. Second, the user may not have any opinion as to the credibility of a source. Forcing the user to establish an ordering could produce some unexpected results. Third, the purpose of having information about the source of a belief is not necessarily so that the system can believe belief A over belief B. For example, the system proposed in this thesis needs this credibility information to decide whether or not to ***accept*** a belief as a held belief. Source ordering information would be of no use in this situation. However, the most serious disadvantage of the ordering method is that a different ordering would be needed for each topic. For example, Sally could believe that John is more credible than Mary with respect to matters of science but Mary is more credible than John with respect to matters of law.

Another way of determining the reader's confidence in the source's competence relative to the subject of the belief is to examine the beliefs about the source in the reader model. Some of the beliefs about the source might give some evidence as to whether or not the reader has confidence in the source. Using a semantic dictionary, some of the meaning of the belief can be determined. If the source is talking about medical matters and the reader has a belief that the source is a lawyer, the system could determine that the reader has low or neutral confidence in the source with respect to the subject of the belief. The system could also determine the reader's confidence is the source as a source regardless of the subject of the belief. For example, the reader may have the belief that the source is a right-wing republican and another belief that all republicans lie. The system would then have to conclude, based on these beliefs, that the reader's confidence in the source is low. In this example, the reader's beliefs concerning the source do not directly correlate to the reader's confidence in the source's competence relative to the subject of the belief. Rather, they may bear more on the reader's confidence in the source as a source. If the reader has no prior beliefs about the source or if none of his prior beliefs about the source are relevant, i.e. do not give any indication as to the reader's confidence in the source, then it is assumed that the reader's confidence in the source is neutral.

This method of determining the credibility of the source is similar to the ordering method in that it relies on the reader's beliefs about the sources however, it is different in that it is more fine-grained. It allows for the reader to have confidence in the source with regards to one topic and not to another. It also allows for the system to determine if the reader has more confidence in one source than another without requiring the reader to state that preference. And most importantly, since the model of the reader is dynamic, the beliefs that the reader model contains about the source will naturally change overtime without additional work on the part of the reader.

The evaluation of the reader's confidence in the source's competence is highly dependent upon the reader's current set of beliefs. This set of beliefs is very dynamic and as such, the evaluation must occur on an as needed basis. No pre-determination can be accomplished and the evaluation is only valid within that particular nesting of environments.

6.4.2 Reader's Confidence in the Reporter

In much the same manner that a reader may have beliefs concerning the source, the reader may have beliefs concerning the reporter. The reader might know a great deal about the reporter's political leanings, past work experience, education, etc. These beliefs could significantly impact on the reader's confidence in the reporter's competence relative to the subject of the belief as well as on the reader's confidence in the reporter's competence as a reporter. For instance, the reader could know that the reporter was educated at Yale University's School of Journalism and has been working as a White House reporter for 10 years. When examining these beliefs, the system could determine that the reader has high confidence in the reporter's competence as a reporter.

However, further examination of the reader's beliefs might yield the finding that the reader also has the belief that the reporter has over the years favoured the democratic side in all his reporting. The system might also have the belief that the reader is a republican. Given a newspaper article about the democratic president's foreign policies, the system would likely conclude that although the reader has confidence in the reporter's competence as a reporter, he does not have confidence in the reporter's competence relative to reporting on the democratic president's foreign policies.

In the case that the reader has no prior beliefs about the reporter or that none of the prior beliefs about the reporter are relevant to the determination of the reader's confidence in the reporter, it is assumed that the reader's confidence in the reporter is neutral.

Like the evaluation of the reader's confidence in the source, the reader's confidence in the reporter is highly dependent on the reader's current set of beliefs. The reader's confidence can not be predetermined and as such must be evaluated when needed and is only valid within that particular nesting of environments.

6.4.3 Reporter's Confidence in the Source

The reporter has the opportunity to encode information concerning his confidence in the source through his choice of words. By analysing the ***lexical semantics of the source*** as discussed in Section 3.2.1, it is possible to determine what if any information about the source has been encoded by the reporter.

The source's identity, expertise, experience and relevance to the situation can provide insight into the situation of the original utterance and help the reader to determine the reliability of the source. The reporter can encode his own assessment of the source, thus giving the reader clues as to how the reporter wants the reader to evaluate the source and interpret the reported speech. This is done through the lexical realisation of the source. How the reporter refers to the source indicates the manner in which the reader is to interpret the reported speech. For instance, in order to bolster a reader's confidence in the source's competence with respect to medicine, a reporter might introduce the source as 'Dr.X, Chief of Staff at Y Hospital and lecturer at Z University's School of Medicine'. This juxtaposition of qualitative noun phrases can be stretched by adding more and more supportive information about the source as required to achieve the desired confidence level.

Contrarily, when a reporter does not provide any supportive information about the source or, indeed, even the source's name, title or connection to the subject matter, it implies a lack of confidence in the source's competence. If the reader has no previous experience with the source, he will most likely rely on the reporter's indicated evaluations.

The evaluation of the reporter's confidence in the source does not depend upon any particular nesting of environments. It must be done during an earlier phase of evidential analysis where the reporter's encodings are extracted from the news article since the lexical realisation of the source is not retained after text analysis is complete. The encodings are valid regardless of whom the reader may be.

6.4.4 Reporter's Confidence in the Reported Information

The *lexical semantics* of the reporting verb, as was discussed in Section 3.2.2, plays an important role in the task of interpreting the original utterance. The reporting verb partly specifies the situation of the original speech utterance, the original context. It can be used to partly specify the manner, intention, and force of the original utterance. The particular verb used by the reporter provides an evaluation of the reporter's belief in the credibility of the information.

An examination of the reporting verb and the determination of its semantic dimensions (see Section 3.4), can allow the reader to assess the reporter's level of confidence in the credibility or reliability of the reported information. Some reporting verbs such as *say* or *state* are very neutral in terms of reliability, i.e., the reporter is not expressing an opinion as to the reliability of the belief. Reporting verbs such as *concede* and *affirm* denote high reliability. On the other hand, the reporting verb *claim* indicates low reliability.

Section 3.3 described how to use these and other factors to evaluate whether or not a belief is reliable. Evidential analysis contends that the complement sentence can be considered true if the circumstantial information is determined to be reliable. Evidential analysis will yield some evaluation of the reliability of the reported speech.

Like the evaluation of the reporter's confidence in the source, the evaluation of the reporter's confidence in the belief (or reported information) does not depend upon any particular nesting of environments. It must be done prior in an earlier phase of evidential analysis where the reporter's encodings are extracted from the newspaper text since the reporting verb is not retained after text analysis is complete. The encodings are valid regardless of whom the reader may be.

6.4.5 Reader's Previous Beliefs about Topic of Belief

The reader's previous beliefs about the topic of the belief are an important factor for the belief heuristic. People tend to believe things more readily if they already have similar beliefs than if they have opposing beliefs. For instance, John reads that university graduates find a job within six months of graduation. However, John, a university graduate himself, has been looking for a job since his graduation 2 years ago. He may very well have the belief that university graduates do not find a job within six months of graduation. In this case, John's belief opposes the proposition that he has read. It will be more difficult to convince him of the truth of the proposition from the article than it would be if John had found a job within six months of graduation and had the belief that university graduates find a job within six months of graduation.

From the example, it is clear that a belief heuristic must take opposing beliefs into account when determining the reliability of a belief. The belief heuristic will determine when and if a potential belief can become a held belief. When a potential belief becomes a held belief, it means that the reader has accepted the potential belief as true. Taking this one step further, to simulate the fact that people are more difficult to convince if they have opposing beliefs, the belief heuristic could make it more difficult for a potential belief to become a held belief when the reader has opposing beliefs, hence the name *opposing belief criterion*. This can be accomplished in many ways. Section 6.4.6 will describe the method used in this thesis.

For the purpose of this thesis, the reader's previous supporting beliefs can only be of one form: a previous exposure to the same belief. Previous supporting beliefs add to the reliability of the belief by means of the ***source list***. The source list is described in Section 6.5.1. A more sophisticated system could take into consideration such situations as whether the reader holds a previous belief that is (a) a necessary precondition, (b) a sufficient precondition, or (c) a consequence of the to-be-acquired belief. A system with a large lexicon and sophisticated reasoning would also be able to recognise that the belief dry(towel) and belief ¬wet(towel) are supporting beliefs. The advanced reasoning needed to handle these types of supporting beliefs is known in the automatic reasoning community to be very difficult. As such, these types of supporting beliefs are beyond the scope of this thesis.

6.4.6 Belief Heuristic

The previous subsections described the need to evaluate the four confidence criteria: the reader's confidence in the source, the reader's confidence in the reporter, the reporter's confidence in the source, and the reporter's confidence in the reported information as well as the opposing belief criterion in order to determine the reliability of a belief.

The manner in which these factors are combined to form a comprehensive quantitative evaluation of the reliability of a belief is a ***belief heuristic***. The belief heuristic will be used to determine when and if a potential belief can be transformed into a held belief. It will do so by determining whether or not the belief's reliability is above a certain threshold.

I have chosen to call this a heuristic because many methods for determining reliability quantitatively are possible. For instance, the four confidence criteria could be evaluated to a numeric value. The four criteria could also be given individual weightings. A formula would then need to be composed which would take into consideration the relative weightings of the criteria.

Two major dependencies between the criteria must be incorporated into any formula for determining the reliability of a belief. The first dependency is between the reader's confidence in the source and the reporter's confidence in the source. The formula should recognise that the reader's confidence in the source is more important. It should have more weight in the formula. The second dependency is between the reader's confidence in the reporter and the two criteria based on the reporter's encodings in the text: the reporter's confidence in the source and the reporter's confidence in the reported information. If the reader has low confidence in the reporter, then the reporter's evaluations should have less weight in the formula. On the other hand, if the reader has high confidence in the reporter, then his evaluations should have more weight.

The result of the evaluation of the reliability of a belief could be a numeric value that when above a certain threshold would indicate that the belief is reliable enough to become a held belief.

As previously mentioned, the belief heuristic must also take into consideration the opposing belief criterion. It must incorporate a method for simulating the fact that a reader will not accept a belief as true as easily if he has opposing beliefs. The method used in this thesis is to have two reliability thresholds. If the reader does not have opposing beliefs, the potential belief will become a held belief if the reliability of the belief as evaluated by the belief heuristic is above the lower reliability threshold. If the reader does have opposing beliefs, then the higher reliability threshold is used.

The belief heuristic also has to provide a mechanism by which the belief's reliability can increase or decrease. For example, sometimes we read something that we do not initially believe. However, if we are repeatedly exposed to this same information, our confidence in its reliability increases. Any method for quantifying belief reliability must take this phenomenon into consideration.

There has been considerable study done in the area of the quantification of reliability. Researchers have developed such methods as applying probabilities, binary quantification (i.e. reliable, not-reliable), tri-level quantification (i.e. low, medium, high) numeric quantification and ranking measures [Bacchus *et al.*, 1994, Snow, 1994, Weydert, 1994]. It is beyond the scope of this thesis to determine which if any of the available methods is sufficient to adequately quantify reliability. However, Chapter 7 will discuss one belief heuristic that was used in the implementation of ***Percolator***, a system developed to exercise the percolation mechanism presented in this thesis.

6.5 Tracing the Source

One of the deficiencies of belief percolation as characterised by Wilks, Bien and Ballim [Wilks and Bien, 1983, Ballim, 1992] is that by virtue of being a side-effect of the belief ascription process, there is no mechanism for keeping track of the source of a belief. When nested environments are decomposed, beliefs that were attributed or ascribed into inner environments can appear as beliefs in outer environments. In the case of ascribed beliefs, the source of the belief can be found by traversing the nestings. However, in the case of percolated beliefs, the nestings have been decomposed. Being able to trace the source of a belief is especially important for belief percolation because it is "a method in which beliefs propagate about a belief system in a way not necessarily intended by any believer or participant, but which follows as a side effect of our principal algorithm" [Wilks and Bien, 1983, p.96]. The believers have no reason or evidence for believing a percolated belief. By evaluating the sources, the believer might find a reason or evidence for acquiring the belief.

6.5.1 Source List

In order for a belief percolation mechanism to be effective, it must incorporate a component that can trace or keep track of the source of a belief. A variety of methods for tracing or keeping track of a source can be devised. Johnson and Shapiro [Johnson and Shapiro, 2000] encode source information in a meta-proposition such that the source information is a belief about a belief. Source information in this form can be added, removed or changed at any time. One disadvantage of this technique is that it can become quite unwieldy. In fact, the implementation of this method, a system called SNeBR could only handle a single source at a time.

This thesis will propose one possible method, a ***source list***, that has no limit as to the number of sources that it can handle at a time. In addition, since it is separate from the proposition, it does not complicate any reasoning activities based on the proposition.

A source list is associated with each belief in an environment. The source list is a list that contains frame-like structures. Each frame represents one exposure to the belief. It contains a sub-list of the names of the sources of a belief. It also contains a slot for the name of the reporter reporting the belief. The frame slots are filled in as information becomes available.

Source Name List:	J. Smith, Officials at City Hall
Reporter Name:	Chris Thompson

Figure 16: Example of a Source List Frame

Figure 16 shows an example of a frame. In this example, the source name list contains the names of two sources: J. Smith and Officials at City Hall. The reporter name slot contains the name of the reporter Chris Thompson. With each successive exposure to the belief, a new frame would be added to the source list.

6.5.2 Building the Source List

The source list is constructed by adding a frame each time the reader is exposed to the associated belief. The **Source Name List** is created by adding to the list a string identifying the owner of the viewpoint of origin after each percolation. If the owner of the viewpoint of origin is the reporter, then the name of the reporter is put in the **Reporter Name** slot and not the **Source Name List**. After successive percolations from inner environments to outer environments, the **Source Name List** and the **Reporter Name** slot would contain all the necessary information to determine the origin of the belief. Figure 17 demonstrates how such a source list is constructed as the result of successive percolations. A simplified form of belief diagrams is used in which only the viewpoint environments are shown. This is done to clarify the diagrams since at this point, the tracing mechanism and not the actual state of the belief environments is the concern.

As shown in the first part of Figure 17, the belief disease(Hodgkins) appears only in the inner environment of Dr. R. Wolf. In this case, the source list is empty. It is not known how Dr. Wolf acquired the belief nor if he accepts it as true, in fact, he could be lying. Therefore, it is taken to be a belief in this viewpoint although no claim is made as to whether it is a *held* belief or a *potential* belief.

The reporter's viewpoint contains the belief expert(Dr. R. Wolf). This belief represents the lexical realisation of the source as described in Section 3.2.1. Transforming the source description into a belief and placing it in a topic environment for the source in the reporter's viewpoint is one way of representing in the nesting the reporter's encoding of his confidence in the source. As previously discussed, topic environments are not shown in this diagram so the reporter's belief about the source is placed directly in the reporter's viewpoint.

The reader's viewpoint contains the belief medical_reporter(S. Johnson). This is a previous belief that the reader has about the reporter. It would normally appear in a topic environment for S. Johnson. This belief is used to determine the reader's confidence in the reporter as described in Section 6.4.1. In this example, the reader believes that the reporter is a medical reporter which indicates that the reporter should be competent with respect to the subject of the belief.

In the second part of Figure 17, the belief is percolated into the viewpoint for S. Johnson the reporter. An important point must be made here: since everything an agent hears or reads is potentially a belief, the belief is at least a potential belief for the reporter and may, in fact, be a held belief. However, since there is no evidence to support a claim, no claim is being made as to whether the reporter has accepted this belief as true or not. It is simply a belief neither held nor potential. For purposes of simplicity, the belief is percolated directly and is not modified in any way. The Source Name List is updated by adding the name of the owner of the origin viewpoint: Dr. R. Wolf.

In the last part of Figure 17, the belief is percolated as a *potential* belief into the environment of the Reader. A string is put in the Reporter Name slot identifying the reporter: S. Johnson. Again, for reasons of simplicity, the belief is percolated in an unmodified form. The source list now contains all the information needed to identify the two viewpoints through which the belief has passed.

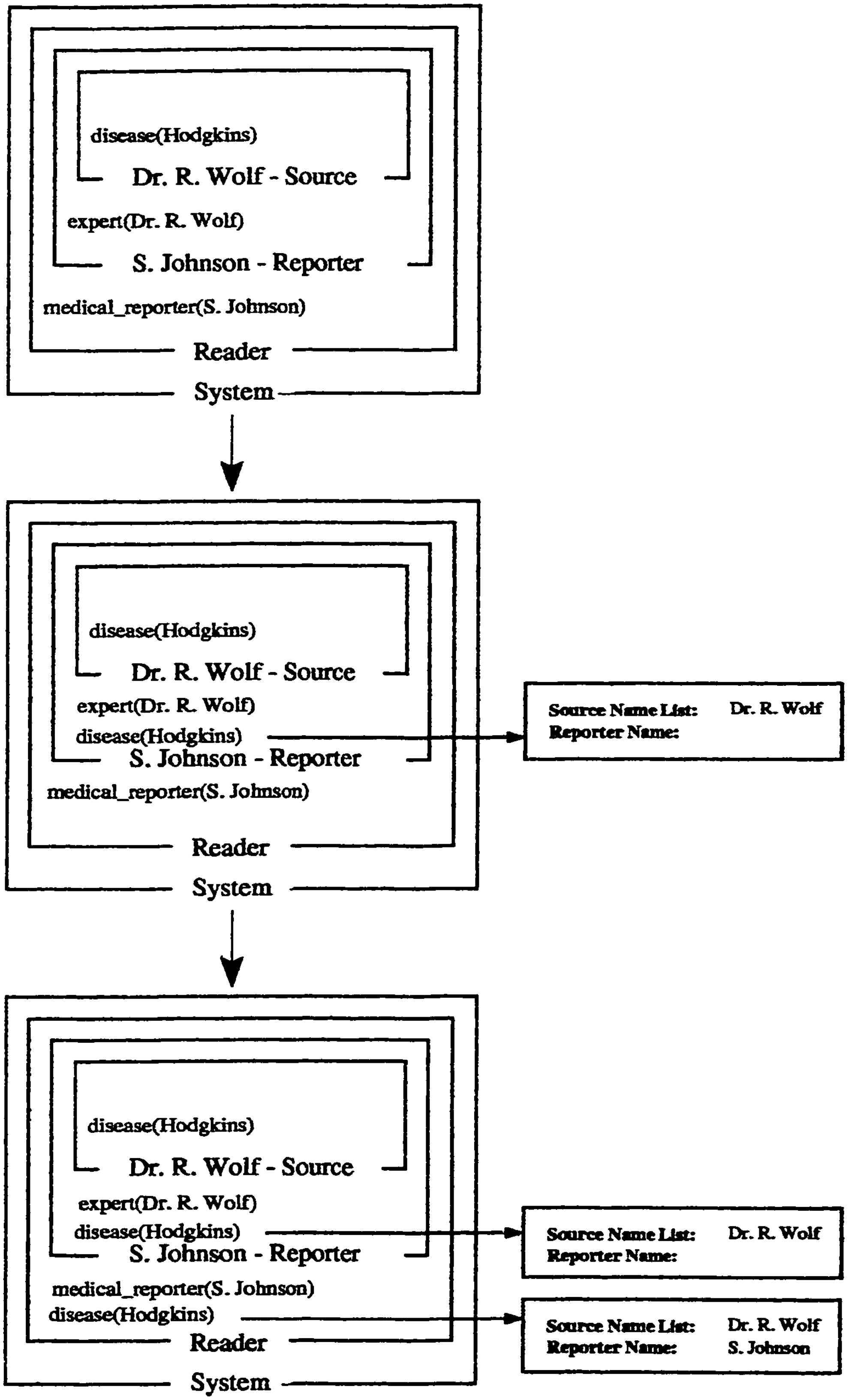

Figure 17: Source List After Successive Percolations

6.5.3 The List After Decomposition

As explained, when the nestings are no longer needed, they are decomposed and only the relevant environments are retained. Some implementations might choose to keep only the viewpoint of the reader while others might see an advantage to also keeping the viewpoints of the reporter and the source. It could be useful information for determining the reliability of the reporter and the source in later evaluations. The viewpoint that is most important for user modelling is that of the reader. Therefore, for the purposes of this thesis, the only viewpoint kept after a decomposition of the nested environments will be that of the reader. Figure 18 depicts the state of the belief set of the reader after decomposition of the nested environments has occurred.

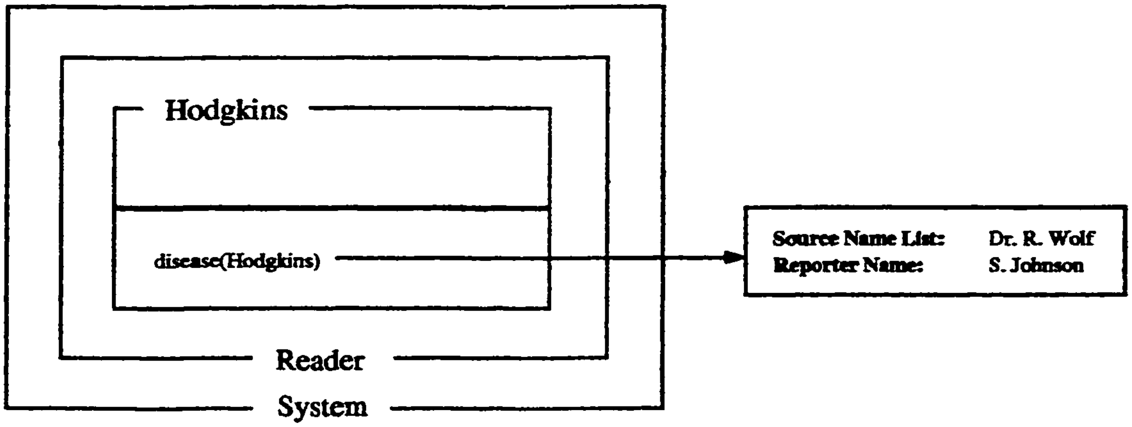

Figure 18: Source List After Decomposition of Environments

The belief that Hodgkins is a disease is now a part of the Reader's viewpoint contained in a topic environment for Hodgkins. The horizontal line in the Hodgkins topic environment is used to separate the *potential* beliefs from the *held* beliefs. The Hodgkins belief is percolated into the reader's environment for potential beliefs about Hodgkins. Attached to it is the full trace of the sources of this belief. There is now some reason or evidence for believing the belief. By traversing the source list, it can be determined that the source for one exposure to the belief was Dr. R. Wolf and that the information was reported by S. Johnson.

6.5.4 Expanding the Source List

In Section 6.4, the four confidence criteria, crucial for a good reliability heuristic, were discussed. In order for a belief heuristic to be effective, it must be able to account for an increase or decrease in reliability based on the frequency with which the reader reads the information. To do this, the formula used to calculate the overall reliability of a belief needs to have access to the reliability assessments for each of the four confidence criteria.

One method of accomplishing this is to include the reliability assessments as part of the source list. Each frame would then consist of a source name list, a reporter name and the ratings of the four confidence criteria: the reporter's confidence in the source (RepConfSource), the reporter's confidence in the reported information (RepConfInfo), the reader's confidence in the source (ReadConfSource), and the reader's confidence in the reporter (ReadConfRep). The ratings of the four confidence criteria can have a value of HIGH, NEUTRAL, or LOW. Figure 19 is an updated version of Figure 16.

Source Name List:	J. Smith, Officials at City Hall
Reporter Name:	Chris Thompson
RepConfSource:	HIGH
RepConfInfo:	NEUTRAL
ReadConfSource:	LOW
ReadConfRep:	NEUTRAL

Figure 19: Expanded Source List Frame

In this example, the frame is completed as follows: the Source Name List contains the names of two sources: J. Smith and Officials at City Hall; the Reporter Name slot contains the name of the reporter, Chris Thompson. The values in the slots for the four confidence criteria require more explanation.

The RepConfSource slot contains the value HIGH indicating that the reporter, Chris Thompson has high confidence in the combined reliability of the sources, J. Smith and Officials at City Hall.

The RepConfInfo slot contains the value NEUTRAL indicating that the reporter, Chris Thompson has neutral confidence in the information he is reporting[1].

The ReadConfSource contains the value LOW. This indicates that the reader has prior beliefs about the sources such that the reader has low confidence in the combined reliability of the sources.

The ReadConfRep slot contains the value NEUTRAL. This indicates one of two situations: (1) the reader has no prior beliefs about the reporter that are relevant to his evaluation of the reporter's reliability (i.e., the reader has no reason to have confidence in the reporter and no reason not to have confidence in the reporter), or (2) the reader has prior beliefs about the reporter that, together, evaluate to a neutral rating (i.e., the reader has some beliefs that indicate that the reader should have confidence in the reporter and some beliefs that indicate he shouldn't have confidence in the reporter).

[1]This is the belief with which the frame is associated. A neutral rating indicates that the reporter has not encoded any reason for the reader to either believe or not to believe the proposition of the belief.

Figure 20 illustrates how the source list frame containing the source information and the reliability assessments is completed during the percolation of a belief.

Initially, there is no source list since the belief was not percolated into Dr. R. Wolf's viewpoint. As previously mentioned, the belief is not assumed to be either a held belief or a potential belief. There is not enough evidence to support either claim. The belief is then percolated into the viewpoint of S. Johnson, the reporter. Again, no claim is made as to the status of the belief. The name of the source is added to the **Source Name List**.

In the previous section, I suggested that the lexicalisation of the source could be stored as a belief in the reporter's viewpoint. The belief could then be used to evaluate the reporter's confidence in the source. Another strategy is to evaluate the reporter's confidence in the source during text analysis since all the necessary information is in the text. The result of the evidential analysis could be stored in the source list and no topic environment for the source would be needed in the reporter's viewpoint. This is the approach taken in the following diagrams.

Thus, the **RepConfSource** slot is filled with the assessment of the reporter's confidence in the source. This assessment is based on the lexical realisation of the source which is how the reporter refers to the source. The reporter encodes his confidence in the reliability of the source in his choice of lexical realisations. This was discussed in Section 6.4.3. The **RepConfInfo** slot is filled with the assessment of the reporter's confidence in the reported information. The assessment is based on the reporter's choice of reporting verbs which is used to encode his confidence in the reported information as was discussed in Section 6.4.4.

The belief is then percolated into the viewpoint of the reader as a potential belief. The **Reporter Name** slot is filled with the name of the reporter. The **ReadConfRep** slot is filled with the assessment of the reader's confidence in the reporter. This assessment is based on the reader's previous beliefs about the reporter as described in Section 6.4.2. In this example, the reader has a previous belief about the reporter indicating that the reporter should be competent with respect to medical issues. The **ReadConfSource** slot is filled with the assessment of the reader's confidence in the source. It, too, is based on the reader's previous beliefs in the source as discussed in Section 6.4.1.

The entire source list is maintained in order to provide traceability to the original source and to all viewpoints through which the belief has passed on its path to the reader's viewpoint.

This combination of source list and belief reliability assessment is easily expandable to enable the tracing of the sources through several readings of the information. The source list as shown in Figure 20 can be expanded to add a second or many other additional source lists by adding a new frame to the list. Each frame represents one exposure to the belief. For instance, if the reader reads this same information, that Hodgkins is a disease, the path of this second reading of the information will be combined with that of the first reading to form an overall source list. This is illustrated in Figure 21.

The information from the different readings of the belief are contained within frames. The first frame contains the source and reporter for the first reading of the belief, along with the assessments for the four confidence criteria. The second frame contains the source and reporter for the second reading of the belief along with the associated assessments. The source list can be extended as often as necessary. The sources for each reading can be traced independently. The assessments pertaining to each particular reading are maintained as long as necessary.

The use of a source list to provide evidence for a belief presents three important advantages. First, the source list allows hearsay to be passed on as such. Second, the source list allows for a reevaluation of the confidence criteria in light of new information about the source or the reporter. If a system is expanded to maintain a source list for held beliefs as well, then the source list can be used to reevaluate the confidence criteria for the beliefs involved in a contradiction to determine which contradictory belief should be retained. Thirdly, the source list could be used to retain the original strings used for the reporting verb and the lexicalisation of the source instead of the current three-valued assessments. In this way, when hearsay is percolated or attributed to another viewpoint, evidential analysis can be performed again. The evaluation of the reporter's confidence in the source and the reporter's confidence in the reported information in the target viewpoint can then differ from the evaluation of the original reader.

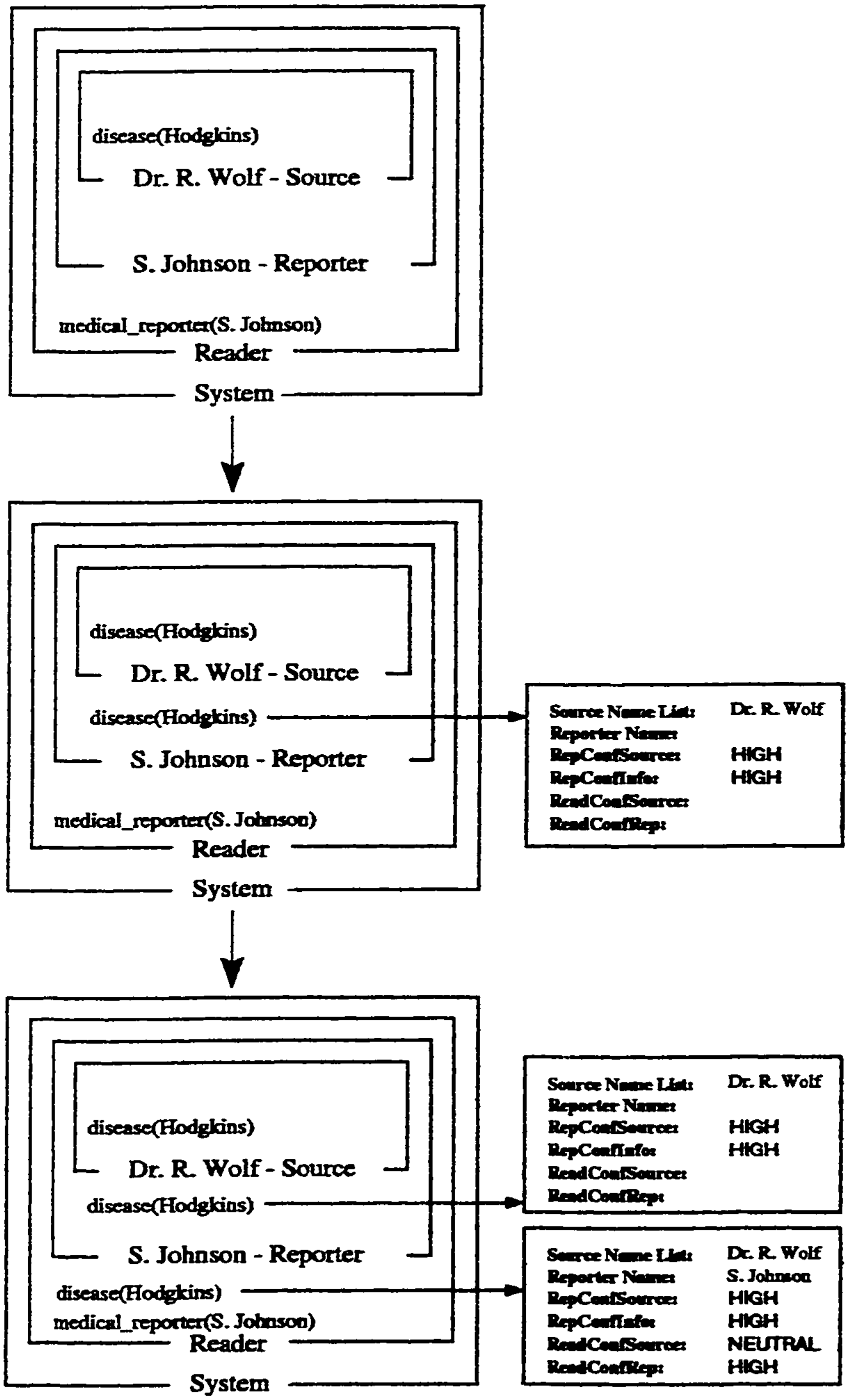

Figure 20: Source List Containing Reliability Assessments

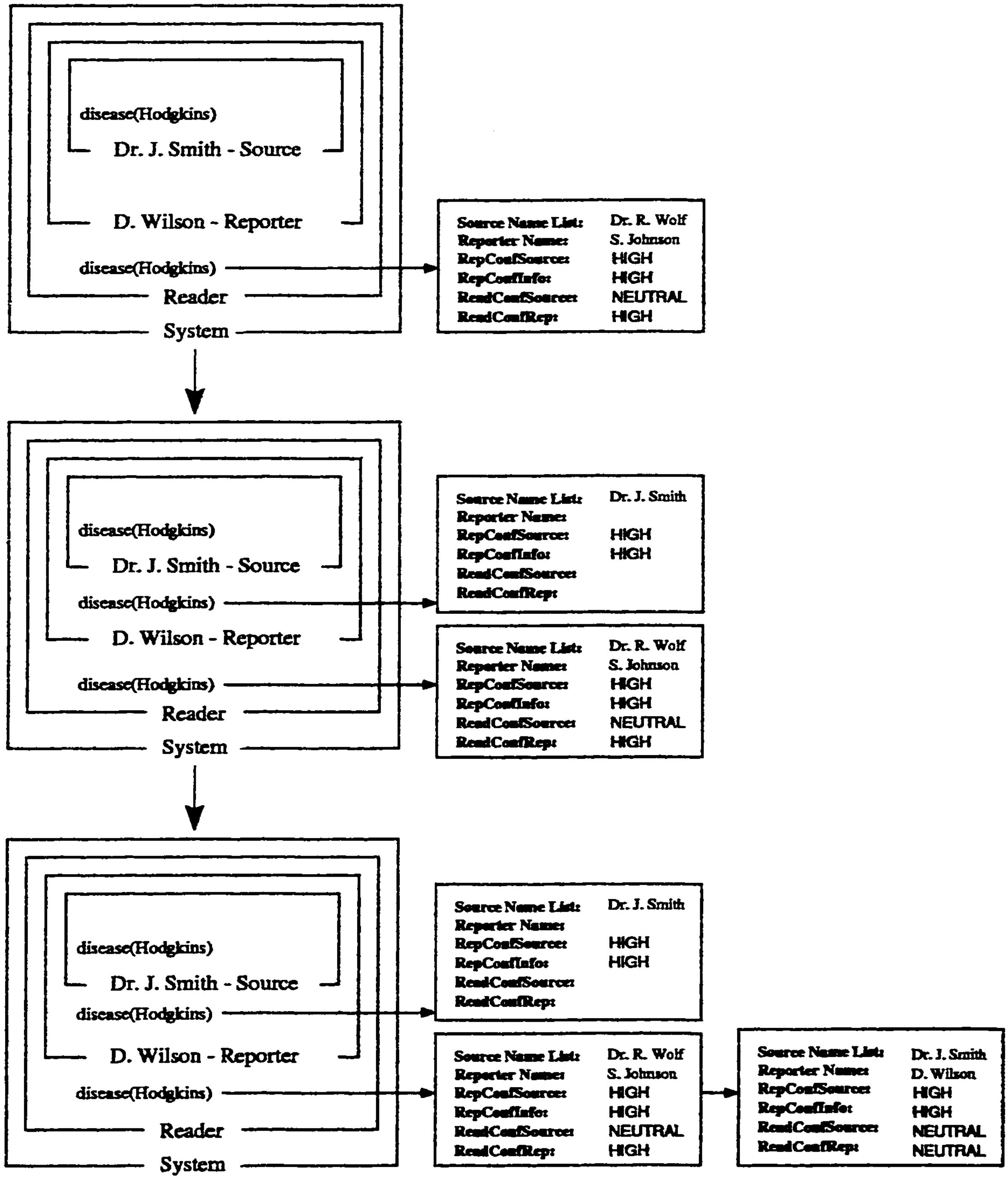

Figure 21: Source List After Two Readings

6.6 Belief Promotion

As discussed earlier, the advantage of distinguishing between held and potential beliefs is that it allows the agent to hold contradictory beliefs, with the stipulation that both beliefs can not be held beliefs, until such a time that one of the contradictory beliefs is removed from the viewpoint. It is common for a reader to have a belief about something and be exposed to a contradictory belief. For a certain period of time, the reader will hold on to the held belief while also keeping the new belief as a potential belief. After repeated exposures to the potential belief, the reader will decide if the belief is of a high enough reliability to override the held belief that it contradicts. If so, ***belief promotion*** will occur. ***Belief promotion*** is the mechanism by which a ***potential*** belief becomes a ***held*** belief. In a system, such as the one proposed in this thesis, the belief heuristic will replace the reader in the decision-making process.

In this case, the reader has opposing beliefs, so based on the ***opposing belief criterion***, the belief heuristic will use a higher reliability threshold to determine if the belief has enough reliability to be promoted. If the belief heuristic determines that the reliability of the ***potential*** belief is above the higher reliability threshold, the held belief is discarded, the belief maintenance mechanism is invoked and the ***potential*** belief is promoted to a ***held*** belief. Figures 22, 23, 24, and 25 provide an example of belief promotion when the reader has an opposing held belief.

As an example, consider the case of Mary, a democrat, who reads that a certain democratic president vows that he has not had sexual relations with a certain White House intern. Given Mary's political leanings and her support of the President, Mary believes the information. Her viewpoint comes to contain the held belief not_sexual_relations(president,intern) as shown in Figure 22.

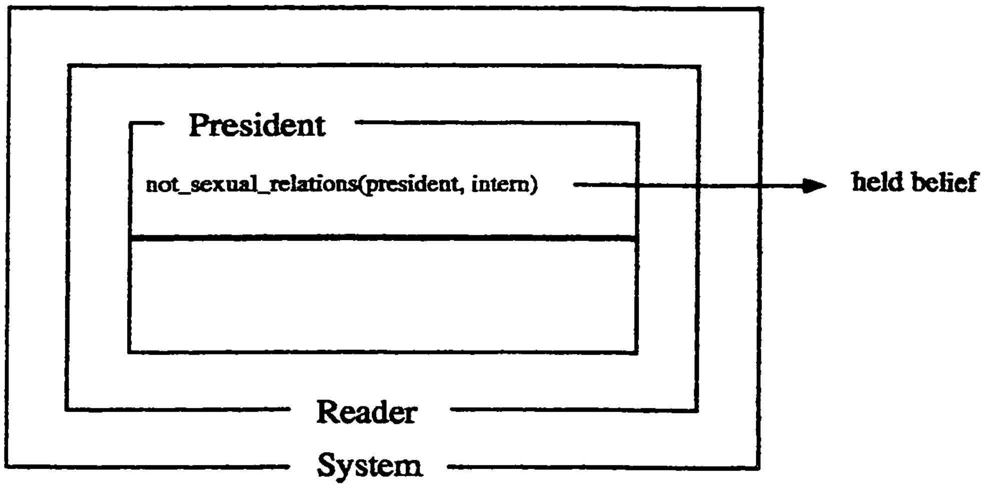

Figure 22: Reversal of Belief - Initial State of Viewpoint

However, a while later, Mary is exposed to the belief that this president had, in fact, had sexual relations of a sort with an intern. The belief is then percolated into the topic environment of Mary's viewpoint as a ***potential*** belief. Evidential analysis is performed on the text and, as described in Section 6.4.3, the reporter's confidence in the source is evaluated. For this example, it will be assumed that the reporter's confidence in the source is determined to be HIGH. The reporter's confidence in the reported information, as described in Section 6.4.4, is also evaluated by evidential analysis. Again, for the purposes of the example, it is assumed that the reporter's confidence in the reported information is determined to be HIGH. It is also assumed that Mary has no previous relevant beliefs about the source or the reporter. So, the ratings for Mary's confidence in the source and the reader's confidence in the reporter are both NEUTRAL. The evaluation results are added to the source list as shown in Figure 23.

The source list contains one frame representing one exposure to the belief. At this point, Mary's viewpoint contains two contradictory beliefs: the belief that the president didn't have sexual relations with the intern and the belief that the president did have sexual relations with the intern. However, this situation poses no problems for Mary or for the system since the belief that the president did have sexual relations with the intern is only a potential belief and does not participate in any belief ascription activities. After each exposure, the belief heuristic as discussed in Section 6.4.6 combines the four confidence criteria and the opposing belief criterion to determine if the potential belief is reliable enough to be promoted. In this example, the reader has a previous opposing belief, so the belief heuristic will use a higher threshold. The reporter's ratings are HIGH and the reader's ratings are NEUTRAL. However, the reader's evaluations are more important than the reporter's. A belief heuristic should recognise this and determine that the belief has not yet acquired enough reliability to be promoted.

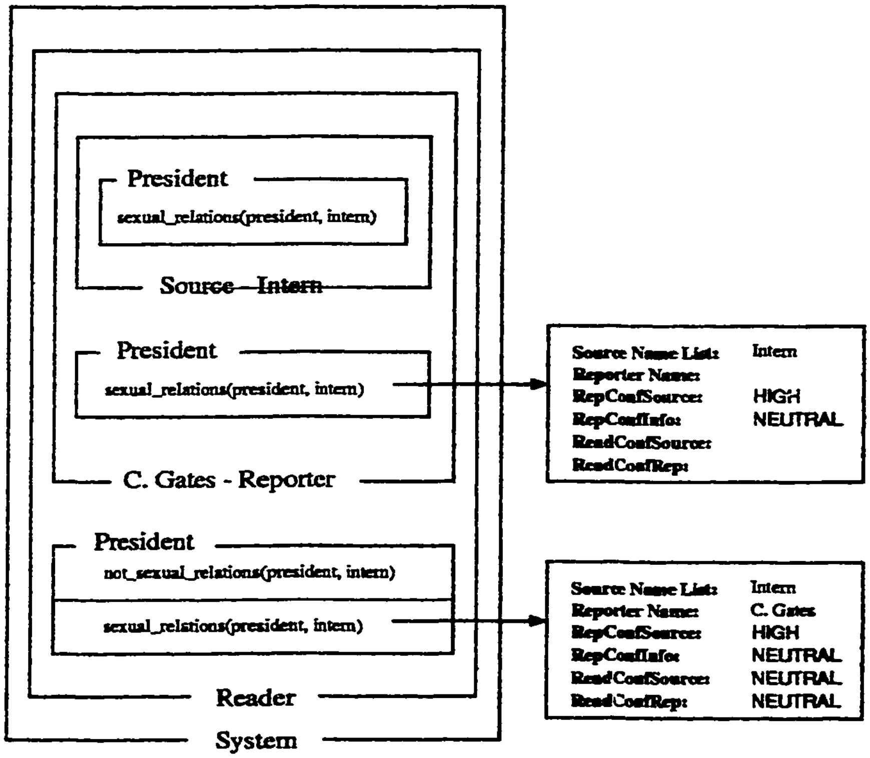

Figure 23: Reversal of Belief - State after First Exposure to Contradictory Belief

Later on, Mary is again exposed to the belief that the president had sexual relations with the intern. Evidential analysis of the text reveals that the reporter's confidence in the source is HIGH and the reporter's confidence in the reported information is also HIGH. Again, to simplify the example, the reader has no previous relevant beliefs about the reporter and the source. So, the reader's confidence in the source and the reader's confidence in the reporter are determined to be NEUTRAL. As can be seen in Figure 24, Mary's viewpoint contains the held belief that the president did not have sexual relations with the intern and the potential belief that the president did have sexual relations with the intern along with a source list now containing two frames representing each exposure to the belief.

The belief heuristic must determine if the belief has acquired enough reliability to be promoted to a held belief. The belief heuristic calculates the reliability of the first exposure to the belief based on the first frame and the reliability of the second exposure based on the second frame. It combines the two reliability assessments to form an overall reliability assessment and compares it to a certain reliability threshold. Since, in this case, the reader has previous opposing beliefs, the opposing belief criterion dictates that the reliability threshold should reflect the fact that the reader is harder to convince of the truth of the belief. For the purposes of the example, it is assumed that the overall reliability assessment exceeds the reliability threshold.

So, belief promotion will occur. The held belief that the president did not have sexual relations with the intern is discarded. The potential belief that the president did have sexual relations with the intern is promoted to a held belief. At this point, the source list can be removed if it is no longer needed. This decision will depend on the type of application for which the model is being used since some applications might find it necessary to maintain the source list even for held beliefs.

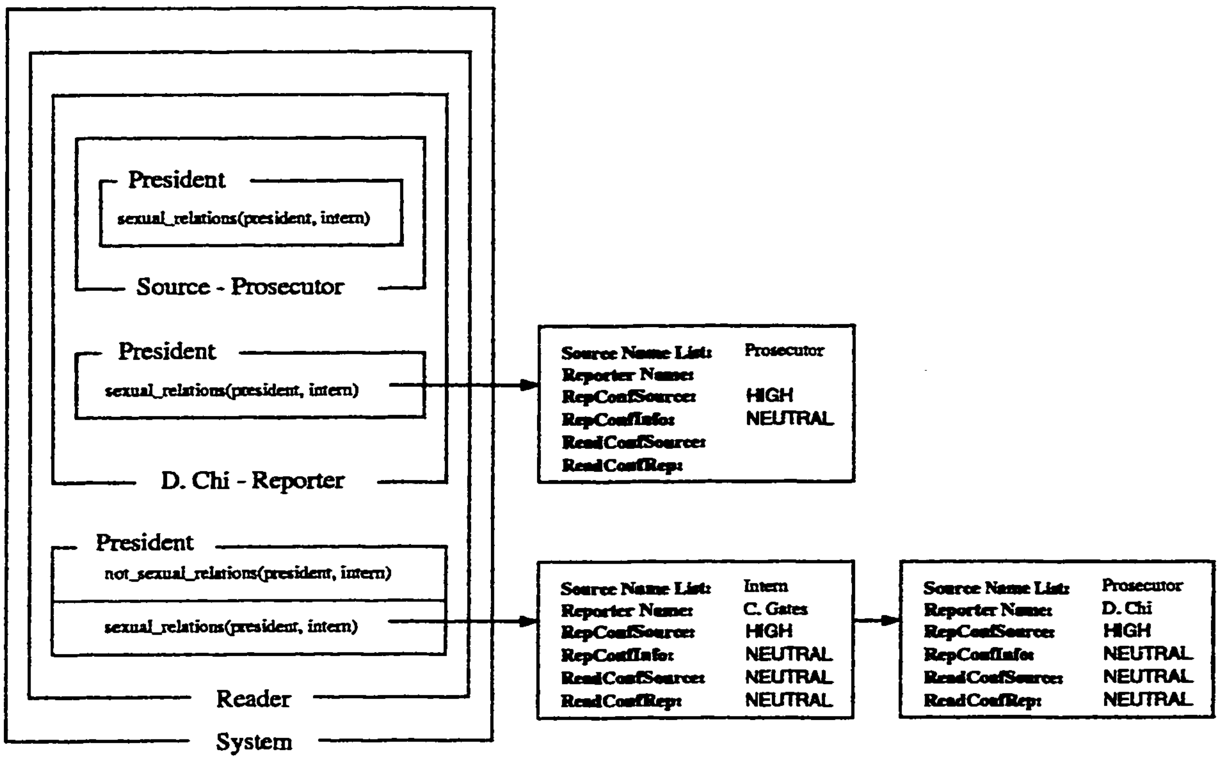

Figure 24: Reversal of Belief - State after Second Exposure to Contradictory Belief

As a final step, the belief maintenance system invokes the belief revision mechanism to remove or revise any beliefs that depended on the discarded belief. The final state of Mary's viewpoint is presented in Figure 25. This type of reversal of beliefs is actually quite common. It is how people correct false beliefs and revise incorrect decisions.

In summary, belief promotion occurs when the belief heuristic determines that the overall reliability of a belief is above a certain threshold. The overall reliability is calculated using the four confidence criteria for each exposure to a belief. The reliability threshold reflects the opposing belief criterion. If the overall reliability of the belief is above the threshold, the associated potential belief is promoted to a held belief. If the reliability of the belief is not above the threshold, the reader must be exposed to the belief again before belief promotion can be successful.

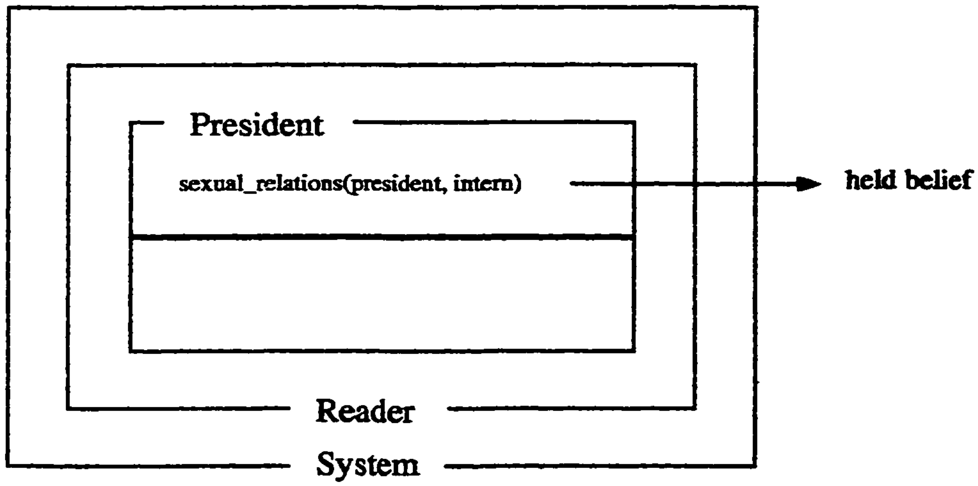

Figure 25: Reversal of Belief - Final State of Viewpoint

Chapter 7

The Percolator System

Percolator is a system I designed to be a proof-of-concept of the ideas presented in this thesis. It is used to build Ballim-type environments as described in Section 4.2. These environments represent the viewpoints of the agents in news articles about the topic of the news article. The most important mechanisms presented in this thesis and implemented in *Percolator* are the percolation mechanism, the decomposition mechanism, and the promotion mechanism guided by a belief heuristic.

As discussed in Section 4.2, viewpoints and topics are defined as types of environments. This can be seen to be analogous to the notion of sub-classing in object-oriented languages. In addition, the type of environment that this system seems most suited for is that of web applications. Given these factors, Java is the ideal language for implementation.

This chapter will give an overview of the classes in *Percolator*. It will explain in detail how the previously mentioned mechanisms were implemented in *Percolator*. The calculation of the four main criteria for belief promotion and their evaluation by the belief heuristic will be described. In addition, the more important design decisions and the reasoning behind them will be discussed.

7.1 Class Hierarchy

The definition of environments and their various types in Section 4.2 was the basis for structuring the classes in *Percolator*. Two of the main object-oriented notions, inheritance and composition are used in this system. Inheritance is represented by the IS-A relationship and composition is represented by the contains or HAS-A relationship. The abstract class **Environment** is the base class. It has two sub-classes: **Viewpoint** and **Topic**. A **Viewpoint** object IS-A **Environment** that contains or HAS-A **Viewpoint** object or a **Topic** object. A **Topic** IS-A **Environment** that HAS-A **Viewpoint** or HAS-A set of **Belief** objects. A **Belief** HAS-A sourceList. A sourceList IS-A list of **SubSource** objects. A **SubSource** HAS-A sourceNameList which contains **String** objects. These class relationships are shown in Figure 26. Each of the classes in *Percolator*, their member data and member methods will be discussed in the following section.

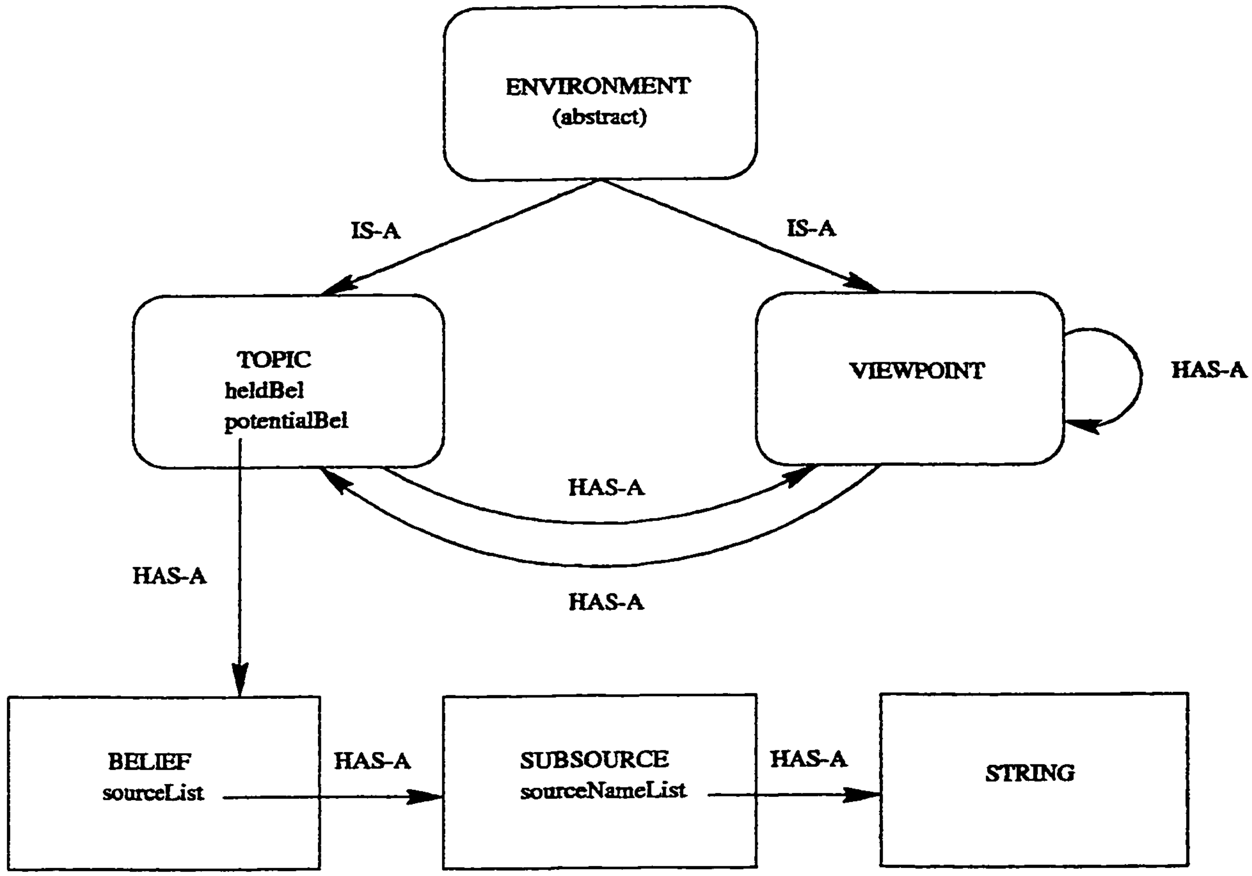

Figure 26: Class Relationships in Percolator

7.2 Class Definitions

In order to build the nesting of environments, there needs to be a link between the parent environment and its child and between the child and it's parent. For example, the *System* viewpoint is always the outer viewpoint. It creates a child viewpoint *Reader*. *System* needs to have a pointer to *Reader* and *Reader* needs to have a pointer to *System*. These links are essential for traversing the environment nesting.

Since this system is meant to be a proof-of-concept and not a commercial system, certain design decisions were made to simplify the implementation. The most important design decision made was to limit the number of child topic objects that a viewpoint object can have. Each viewpoint other than that of the reader can only have one child topic object[1]. The reader, on the other hand, can have many. The main topic object, about the topic under consideration, is the same as for the other viewpoints. However, the reader can have a list of topic objects about the agents in the news articles. This limitation has greatly simplified the traversal of the nestings. Of course, in a commercial system, it might be necessary to expand the structure into that of a multi-noded tree.

[1]This is justified for the purposes of this thesis since a text is processed incrementally. The system processes beliefs about a particular topic one at a time and not beliefs about several different topics at the same time

7.2.1 Environment

The **Environment** class is an abstract base class. It contains data members that all objects of its sub-classes will need. In order to build the nesting of environments, there needs to be a link between the parent environment and its child and between the child and its parent. For example, the ***System*** viewpoint is always the outer viewpoint. It creates a child viewpoint ***Reader***. ***System*** needs to have a pointer to ***Reader*** and ***Reader*** needs to have a pointer to ***System***. These links are essential for traversing the environment nesting. So, **child** and **parent** are included as data members of the **Environment** class. In addition, each environment must have a name. Therefore, **name** is also included as a data member of the **Environment** class. These data members have been given **protected** access which means that only objects of the derived classes have direct access to them (inherit them). Objects of non-derived classes must use the access methods provided by the **Environment** class to set and get the data members.

The class was made **abstract** because an **Environment** object should not be instantiated. An environment must be a viewpoint environment or a topic environment. Making the class abstract enforces this. Below is the class definition for the **Environment** class. It states that **Environment** implements the **Rating** interface (explained in Section 7.2.6) which in the case of *Percolator* simply means that **Environment** and any of its derived classes has access to all the constants (**final static** variables) defined in **Rating**.

```
abstract class Environment implements Rating {
    protected String name;
    protected Environment child;
    protected Environment parent;

    public Environment getChild() { /* ...code omitted...*/ }

    public String getName() { /* ...code omitted...*/ }

    public Environment getParent() { /* ...code omitted...*/ }

    public void printParent() { /* ...code omitted...*/ }

    public void removeChild() { /* ...code omitted...*/ }

    public void removeParent() { /* ...code omitted...*/ }

    public void setParent(Environment e) { /* ...code omitted...*/ }
}
```

7.2.2 Viewpoint

The *Viewpoint* class is a sub-class of the **Environment** class. It inherits the data members and methods of the **Environment** class. It extends the **Environment** class by adding three new data members.

The first new data member, **childTopic**, is a reference to a child **Topic** object. It is used when a topic is created within a viewpoint, i.e. the reader's beliefs about dogs. The **Topic** object named *Dogs* would be a child topic of the **Viewpoint** object named *Reader*. The **child** data member inherited from the base class is used to point to a child **Viewpoint** object.

The second new data member, **parentTopic**, is a reference to a parent topic. In theory, a topic can contain a viewpoint if the agent of the viewpoint is also the subject of the topic. For example, the system might have a topic environment named *Reader* and within that topic environment have a viewpoint environment for *Reader*. This is not used in *Percolator* but was included for consistency and to simplify extension of the system.

The third new data member, **childTopic2**, is a list of child **Topic** objects. Presently, this list is used only by the Reader's viewpoint. It is used to contain all the topic environments that hold the reader's beliefs about the other agents in the nesting. In addition, the **Viewpoint** class adds various member methods for accessing and manipulating the new data members. Included below is the class definition for the **Viewpoint** class.

```
class Viewpoint extends Environment {
    private Topic childTopic;
    private Topic parentTopic;
    private Collection childTopic2 = new ArrayList(); // should only be
                                                       // used by reader

    Viewpoint(String viewpointName) { /* ...code omitted...*/ }

    public Viewpoint createChildViewpoint(String name) { /* ...code omitted...*/ }

    public Topic createChildTopic(String name) { /* ...code omitted...*/ }

    public Topic createChildTopic2(String name) { /* ...code omitted...*/ }

    public void addBeliefToReaderBase(Belief bel, String belAboutName) {
        /* ...code omitted...*/ }

    public Collection getBeliefsAbout(String type, String name) {
        /* ...code omitted...*/ }

    public Topic getChildTopic() { /* ...code omitted...*/ }
```

```
    public Collection getChildTopic2() { /* ...code omitted...*/ }

    public Topic getTopicFromTopicList(String name) { /* ...code omitted...*/ }

    public void removeChildTopic() { /* ...code omitted...*/ }

    public void removeChildTopic2() { /* ...code omitted...*/ }

    public void removeParentTopic() { /* ...code omitted...*/ }

    public void setParentTopic(Topic t) { /* ...code omitted...*/ }
}
```

7.2.3 Topic

The **Topic** class is also a sub-class of the **Environment** class. It extends the **Environment** class by adding two data members. The first new data member, **potentialBel**, is a list of all the potential beliefs the agent has about the topic. The second new data member, **heldBel**, is a list of all the held beliefs the agent has about the topic. The elements stored in the lists are **Belief** objects.

In order to access the new private data members, the **Topic** class also adds various member methods for accessing and manipulating the data members. Below is the class definition for the **Topic** class.

```
class Topic extends Environment {
    private Collection potentialBel = new ArrayList();
    private Collection heldBel = new ArrayList();

    Topic(String topicName){ /* ...code omitted...*/ }

    public Viewpoint createChildViewpoint(String name) { /* ...code omitted...*/ }

    public void addbelief(Belief belief, String type) { /* ...code omitted...*/ }

    public void addbelief(Collection beliefs, String type, SubSource ss) {
        /* ...code omitted...*/ }

    public boolean contains(Collection list, Belief newbel) {
        /* ...code omitted...*/ }

    public boolean containsOpposite(Collection list, Belief newbel) {
        /* ...code omitted...*/ }
```

```
    public Belief getBelief(Collection list, Belief newbel) {
        /* ...code omitted...*/ }

    public Collection getBeliefs(String type) { /* ...code omitted...*/ }

    public void printBeliefs() { /* ...code omitted...*/ }

    public void printBeliefs(String type) { /* ...code omitted...*/ }

    public void promoteBelief(Belief potBel, boolean opposing) {
        /* ...code omitted...*/ }

    public void removeBelief(Collection list, Belief bel) {
        /* ...code omitted...*/ }
}
```

7.2.4 Belief

Objects of type **Belief** are the main concern of *Percolator*. These are the objects that are associated with particular **Topic** objects within particular **Viewpoint** objects. All the main mechanisms described in this thesis manipulate **Belief** objects. This class has two data members.

The first, **proposition** is the logical proposition of the belief. For example, if it is written in the news article: *The CEO of Eaton's Co. has conceded that the company has filed for bankruptcy.*, the logical proposition would be of the form: bankrupt(Eaton's Co.).

The second data member, **sourceList**, is a list of **SubSource** objects. Each time the reader is exposed to the same belief, a **SubSource** is added to the **sourceList**. Below is the class definition for the **Belief** class. It states that **Belief** implements the **Rating** interface so **Belief** has access to all the constants defined in **Rating**.

```
class Belief implements Rating {
    private String proposition;
    private Collection sourcelist = new ArrayList();

    Belief(String belief) { /* ...code omitted...*/ }

    Belief(String belief, int repConfSource, int repConfInfo) {
        /* ...code omitted...*/ }

    Belief(String belief, String repName, String sourceName, int repConfSource,
        int repConfInfo) {
        /* ...code omitted...*/ }
```

```
    Belief(String belief, String repName, String source1, String source2,
        int repConfSource, int repConfInfo) {
        /* ...code omitted...*/ }

    Belief(Belief b) { /* ...code omitted...*/ }

    public void addSourceList(Collection list) { /* ...code omitted...*/ }

    public String getOpposite() { /* ...code omitted...*/ }

    public String getProposition() { /* ...code omitted...*/ }

    public int getPropValue() { /* ...code omitted...*/ }

    public Collection getSourceList() { /* ...code omitted...*/ }

    public boolean isSubString(String keyword) { /* ...code omitted...*/ }

    public void printBelief() { /* ...code omitted...*/ }

    public void printSourceList() { /* ...code omitted...*/ }

    public String toRating(int i) { /* ...code omitted...*/ }

    public void updateLastSubSource(SubSource ss, Viewpoint vp) {
        /* ...code omitted...*/ }
}
```

7.2.5 SubSource

As was discussed in the class definition for **Belief**, the **sourceList** is a list of **SubSource** objects. Each **SubSource** in the lists represents an exposure to a particular belief. Each time the reader is exposed to a belief, all the information relative to that exposure is stored in a **SubSource** object. **SubSource** defines six data members.

The first data member, **sourceNameList** is a list of **String** objects each representing the name of one of the sources of the belief for that particular exposure. For example, if the reporter, J. Alexander, is quoting John Smith who says that Suzanne Taylor said the price of gas would increase, then the **sourceNameList** would include two **String** objects, *John Smith* and *Suzanne Taylor*.

The second data member, **reporterName** is an object of the **String** class. It contains the name of the reporter involved in that particular exposure to the belief. In the above example, the name of the reporter, *J. Alexander*, would be stored as a **String** object pointed to by **reporterName**.

The third data member, **repConfSource**, represents the reporter's confidence in the source as indicated by the reporter's lexical realisation of the source (discussed in Section 3.2.1). It is currently being provided as a parameter to the **Belief** constructor method. Its value is HIGH, NEUTRAL or LOW.

The fourth data member, **repConfInfo**, represents the reporter's confidence in the reported information as indicated by the reporter's choice of reporting verbs (discussed in Section 3.2.2). It is currently being provided as a parameter to the **Belief** constructor method. Its value is HIGH, NEUTRAL or LOW.

The fifth data member, **readerConfRep**, represents the reader's confidence in the reporter. It's value is based on all the beliefs that the reader may have about the reporter in **reporterName**. It is calculated each time a **SubSource** is added to the **sourceList**, that is, with each exposure to the belief. It's value is HIGH, NEUTRAL or LOW.

The sixth data member, **readerConfSource**, represents the reader's confidence in the source or sources of the information. It's value is based on the cumulative evaluation of all the beliefs that the reader may have about all of the sources in the **sourceNameList**. It, too, is calculated each time a **SubSource** is added to the **sourceList**, that is, with each exposure to the belief. It's value is HIGH, NEUTRAL or LOW.

The **SubSource** class also contains many member methods for accessing and manipulating the data members. The class definition for the **SubSource** class is included below. It states that **SubSource** implements the **Rating** interface so **SubSource** has access to all the constants defined in **Rating**.

```
class SubSource implements Rating {
    private Collection sourceNameList = new ArrayList();
    private String reporterName;
    private int repConfSource;
    private int repConfInfo;
    private int readerConfRep;
    private int readerConfSource;

    SubSource() {};

    SubSource(SubSource ss) { /* ...code omitted...*/ }

    public void deleteSourceNames() { /* ...code omitted...*/ }

    public void evalReaderConfSource(Viewpoint vp) { /* ...code omitted...*/ }

    public void evalReaderConfRep(Viewpoint vp) { /* ...code omitted...*/ }

    public int getReaderConfRep() { /* ...code omitted...*/ }
```

```
    public int getReaderConfSource() { /* ...code omitted...*/ }

    public int getRepConfInfo() { /* ...code omitted...*/ }

    public int getRepConfSource() { /* ...code omitted...*/ }

    public String getReporterName() { /* ...code omitted...*/ }

    public Collection getSourceName() { /* ...code omitted...*/ }

    public void printSourceNameList() { /* ...code omitted...*/ }

    public void printSourceNameList(Collection list) { /* ...code omitted...*/ }

    public void setReaderConfRep(int  v) { /* ...code omitted...*/ }

    public void setReaderConfSource(int  v) { /* ...code omitted...*/ }

    public void setRepConfInfo(int  v) { /* ...code omitted...*/ }

    public void setRepConfSource(int  v) { /* ...code omitted...*/ }

    public void setReporterName(String name) { /* ...code omitted...*/ }

    public void setSourceName(String name) { /* ...code omitted...*/ }

    public void setSourceName(Collection list) { /* ...code omitted...*/ }
}
```

7.2.6 Rating

Rating is not a class, rather, it is an interface. A Java interface establishes a form for a class. It states all the method names, argument lists and return types. However, it does not permit method bodies. It is a template that can be used to ensure that classes follow a specified format. An interface can also contain data members of primitive types. These must be declared **static** and **final**. In *Percolator* it is used to hold all the constants that will be needed throughout the system. This is a useful technique since Java does not allow for any type of global variable.

Rating contains six data members. The first three data members define the ratings that can be assigned to the four confidence criteria: the reader's confidence in the reporter, the reader's confidence in the source, the reporter's confidence in the source and, the reporter's confidence in the reported information. The values assigned to these ratings have no real use in *Percolator*, it is the name that is useful. For example, it is easier to understand the code when it reads: if (readerConfRep == LOW) than when it reads: if (readerConfRep == -1). It also contains three tab definitions used when formating *Percolator's* output.

```
interface Rating {
    public static final int LOW = -1;
    public static final int NEUTRAL = 0;
    public static final int HIGH = 1;
    public static final String tab9  = "         ";
    public static final String tab11 = "           ";
    public static final String tab13 = "             ";

}
```

7.2.7 Percolator

Percolator is the only **public** class in the system and as such is the only way to access the package externally. A package is a set of classes that are grouped together. In this case, all of the classes previously mentioned are part of the **Percolator** package. **Percolator** is the class that contains the **main()** method. All of the mechanisms previously described in this thesis are member methods of the **Percolator** class as this is the class that is the *Percolator* application. We could easily have another class, say **Ascriptor**, that would provide the methods to ascribe beliefs from one viewpoint to another. It would use all of the same classes as *Percolator* but would constitute a different application.

Percolator has no data members. It's member methods manipulate the objects of the previously defined classes to implement the main mechanisms developed in this thesis: percolation, decomposition and promotion. It is also in this class that the belief heuristic is implemented. *Percolator* also contains various methods for implementing a menu system and for running several demos pertinent to this thesis. Below is the class definition for the **Percolator** class. The main methods implemented in **Percolator** are described in the following sections.

```
public class Percolator implements Rating  {

    public void beliefHeuristic(Viewpoint system) { /* ...code omitted...*/ }

    public float convertRating(int rating) { /* ...code omitted...*/ }

    public void createFullModel(Viewpoint system) { /* ...code omitted...*/ }

    public void createMinimalModel(Viewpoint system) { /* ...code omitted...*/ }
```

```
    public void createSampleTextModel(Viewpoint system) {/* ...code omitted...*/ }

    public void decompose(Viewpoint system) { /* ...code omitted...*/ }

    public int evaluateRatings(int repConfSource, int repConfInfo,
                               int readerConfSource, int readerConfRep) {
        /* ...code omitted...*/ }

    public void fourthExposureFullMod(Viewpoint system) {
        /* ...code omitted...*/ }

    public void fullDemo(Viewpoint system) { /* ...code omitted...*/ }

    public Viewpoint getinnermost(Viewpoint v) { /* ...code omitted...*/ }

    public int getReaderModel() {/* ...code omitted...*/ }

    public static void main(String args[]) { /* ...code omitted...*/ }

    public void menu(Viewpoint system) { /* ...code omitted...*/ }

    public void minDemo(Viewpoint system) { /* ...code omitted...*/ }

    public void percolate(Viewpoint system) { /* ...code omitted...*/ }

    public void printSystem(Viewpoint system) { /* ...code omitted...*/ }

    public void reset(Viewpoint system) { /* ...code omitted...*/ }

    public void sampleTextDemo (Viewpoint system) { /* ...code omitted...*/ }

    public void secondExposureFullMod(Viewpoint system) {
        /* ...code omitted...*/ }

    public void secondExposureMinMod(Viewpoint system) { /* ...code omitted...*/ }

    public void thirdExposureFullMod(Viewpoint system) { /* ...code omitted...*/ }

    public void thirdExposureMinMod(Viewpoint system) { /* ...code omitted...*/ }
}
```

7.3 Percolation Mechanism

Chapter 5.4 introduced the notion of *percolation* as the mechanism by which inner beliefs become outer beliefs. Chapter 6 explained how percolation is part of the process by which a reader of news articles can acquire beliefs. The default percolation rule defined in Section 6.3 is to ***percolate beliefs as potential beliefs when there is evidence to justify doing so.***

The justification in the case of a system modelling readers of news articles, like *Percolator*, is that the reader is reading articles on current topics or events with the intent of gaining information. The sources of the information are supposed to be competent with respect to the information unless expressly contradicted. In order to safeguard the reader's viewpoint against unreliable information or information contradictory to beliefs already held by the reader, beliefs are percolated as potential beliefs.

Once the nesting of the viewpoint environments has been created, the default percolation rule can be applied by calling the member method **percolate(Viewpoint system)** in the **Percolator** class. The percolation algorithm is the following:

```
find the innermost viewpoint
set parent to the parent of this viewpoint
set origin to this viewpoint

while there is a parent
   set origTopic to the child topic of origin
   set destTopic to the child topic of parent.

   create a SubSource object
   if origin is not the reader or the reporter
      add the name of origin to the sourceNameList
   else if origin is the reporter
      set the name of the reporter

   if destTopic is null create a childTopic of the
      same name as origTopic

   if percolating to the reader
      add beliefs to the reader's topic environment as potential
   else
      add beliefs to the destination topic environment

   set origin to parent
   set parent to the parent of the parent
```

The algorithm is really quite simple. First, find the innermost viewpoint. This is the viewpoint of the source of the reported information. Find the parent of this viewpoint. This will be the next source in the chain of sources or it can be the reporter. The beliefs in the topic environment of the innermost viewpoint will be copied to the child topic of its parent viewpoint. This is done for each viewpoint in the nesting except for that of the system. In the case of the reader, the beliefs are percolated as potential beliefs.

As previously mentioned, beliefs are not separated into held versus potential for the other agents in the nesting because the news article does not provide enough evidence to determine whether the belief is a held belief of the agent or not. Furthermore, the goal of *Percolator* is to model the reader's beliefs not those of the other agents.

As an example, consider a news article in which it is stated: *Dr. J. Smith, Research Director at Mount Sinai Hospital has announced that he has found the cure to the common cold.* A belief with the proposition cure(common_cold,Dr.J. Smith) will be created. The reporter has used a positive lexical realisation for the source so a rating of HIGH will also be added to indicate the reporter's confidence in the source. The reporter has also used the reporting verb *to announce* which also indicates high confidence in the reported information (see Section 3.4 for a discussion on the semantic dimensions of reporting verbs). As a consequence, a rating of HIGH will be added to indicate the reporter's confidence in the reported information. Figure 27 is output from *Percolator* showing an initial nesting of environments for the example. It represents the System's view of Mary's view of M. Johnson's view (the reporter) of Dr. J. Smith's view (the source) of his beliefs concerning the common cold. In this example, neither the reader (Mary) nor the reporter (M. Johnson) have any prior beliefs concerning the common cold. In addition, the reader has no prior beliefs about the reporter or the source.

```
System believes
  Mary believes
    M. Johnson believes
      Dr. J. Smith believes the following:

          Beliefs about Common Cold
            cure(common_cold,Dr. J. Smith)
```

Figure 27: State of the System Before Percolation

At this point, the percolation mechanism can be activated. As shown in Figure 28, the belief is percolated from the source's (Dr. J. Smith) topic environment about the common cold to the reporter's (M. Johnson) topic environment about the common cold. A **SubSource** object is now created to represent the new exposure to the belief. The source's name is added to the **sourceNameList**. Then, the belief is percolated from the reporter's topic environment about the common cold to the reader's (Mary) topic environment about the common cold as a potential belief. At this point, the reporter's name is added to the **SubSource** as well as the ratings for the reporter's confidence in the source and the reporter's confidence in the reported information. In addition, since the names of the source and the reporter are now known, the reader's confidence in the source and the reader's confidence in the reporter can now be evaluated and added to the **SubSource**. The System viewpoint contains Mary's viewpoint and no topic environments. Mary's viewpoint contains the newly percolated potential belief along with all the information concerning this exposure to the belief.

A word needs to be said about the manner in which *Percolator* prints out the viewpoints. First, the nestings are printed from the outside in so the first viewpoint shown is that of the System. In Ballim's belief diagrams, the inner environments are the more important environments since belief ascription proceeds from the outer environments to the inner environments. However, in the case of *Percolator*, the more important environments are the outer environments, more specifically that of the reader, since belief percolation proceeds from the inner environments to the outer environments. Second, as the printing algorithm traverses the nestings, the name of each viewpoint traversed is added to a path variable. Whenever a viewpoint contains a topic environment, the entire path is printed. This is done since the printout can be rather large and occupy more than one screen. If the path is not reprinted, it could look as though the inner viewpoints are not nested. An alternative printing algorithm would be to only print the name of the current viewpoint and not the entire path. In this case, the lines "System believes Mary believes would not be printed out before printing out M. Johnson's beliefs about the common cold. Similarly, the lines "System believes Mary believes M. Johnson believes" would not be printed out before printing out Dr. J. Smith's beliefs about the common cold. In this manner, the printout would be exactly equivalent to the modified belief diagrams as described in Section 6.2.

7.4 Decomposition Mechanism

Once percolation of the belief has taken place, the nestings are no longer needed since all the source information and the information needed for the belief heuristic has been stored in the source list of the belief (in a **SubSource** object). *Decomposition* as described by Ballim consists of percolating the beliefs in the inner environment to the outer environment and then removing the nesting of the environments. The beliefs are then placed in a topic environment in the *System's* viewpoint. Decomposition for the purposes of this thesis differs somewhat. In *Percolator*, beliefs are percolated from the topic environment of the innermost viewpoint environment to the reader's topic environment where they are stored as potential beliefs. This is the percolation mechanism.

At this point, the nesting is *decomposed*. All the environments in the nesting, except for those of the reader and the system, are no longer needed, so they are removed. It is important to keep all of the reader's topic environments and all of the reader's beliefs. The percolation that has taken place only represents one exposure to the belief. The reader's topic environments are needed for future exposures.

In a commercial system, the reader's viewpoint and all its topic environments would be kept as long as the reader is a potential user of the system. Since the goal of the system is to build and maintain a model of the reader's beliefs, it would not make sense to reset the model after each use.

Figure 29 continues on with the example presented in Section 7.3. It shows how the viewpoints of

```
System believes
  Mary believes the following:

        POTENTIAL Beliefs about Common Cold
          cure(common_cold,Dr. J. Smith)
            Source 1: Dr. J. Smith
            Reporter: M. Johnson
              Reporter's confidence in the Source(s):   HIGH
              Reporter's confidence in the Information: HIGH
              Reader's confidence in the Source(s):     NEUTRAL
              Reader's confidence in the Reporter:      NEUTRAL


System believes
  Mary believes
    M. Johnson believes the following:

        Beliefs about Common Cold
          cure(common_cold,Dr. J. Smith)
            Source 1: Dr. J. Smith


System believes
  Mary believes
    M. Johnson believes
      Dr. J. Smith believes the following:

        Beliefs about Common Cold
          cure(common_cold,Dr. J. Smith)
```

Figure 28: State of System after Percolation

the reporter, M. Johnson, and the source, Dr. J. Smith, have been removed. The reader's viewpoint remains and the topic environment about the topic under consideration, the common cold in this case, is printed out along with the information about any exposures to the belief.

After decomposition of the nesting of environments has taken place, a ***belief heuristic*** is used to evaluate whether the reader has enough confidence in any of his potential beliefs to make them held beliefs. The ***belief heuristic*** implemented in ***Percolator*** is discussed in Section 7.5.3.

One design decision that has been made with respect to printing out the system is to show only the beliefs of the reader that are relevant to the particular activity. So, when percolation is taking place, the reader's beliefs about the sources and the reporter are relevant so they are printed out. Once the system has been decomposed, the only relevant beliefs are those in the topic environment of the topic under consideration. However, this does not mean that those topic environments have been removed from the reader's viewpoint, they are simply not printed out.

```
System believes
  Mary believes the following:

          POTENTIAL Beliefs about Common Cold
            cure(common_cold,Dr. J. Smith)
              Source 1: Dr. J. Smith
                Reporter: M. Johnson
                Reporter's confidence in the Source(s):     HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):       NEUTRAL
                Reader's confidence in the Reporter:        NEUTRAL
```

Figure 29: State of the System After Decomposition

7.5 Promotion Mechanism

Promotion of a belief refers to the transition a belief makes when it goes from being a potential belief to a held belief. This transition can only occur when certain pre-conditions have been met. In ***Percolator*** it is the ***belief heuristic*** that evaluates the source list (a list of **SubSource** objects each of which represents an exposure to the belief). As discussed in Section 6.4.6, a belief heuristic must take into consideration four confidence criteria as well as the reader's prior beliefs about the topic.

The following sub-sections will describe the four confidence criteria: the reporter's confidence in the source, the reporter's confidence in the reported information, the reader's confidence in the reporter, and the reader's confidence in the source and how they are obtained. It will also discuss a formula that is used to evaluate the confidence criteria and the thresholds used to determine whether or not a potential belief can be promoted to a held belief. The reader's prior beliefs on the topic under consideration are of great importance. The impact of these beliefs on the decision of whether or not to promote a belief will be shown.

7.5.1 Obtaining the Criteria

The belief heuristic needs to take into consideration the reporter's confidence in the source, the reporter's confidence in the reported information, the reader's confidence in the reporter, and the reader's confidence in the source when determining the reliability of a belief.

The first criterion, the reporter's confidence in the source, is obtained from the text analysis component. This component is not a part of ***Percolator***. It is assumed that the news articles will have already passed through such a component and that only the beliefs in the form of propositions as well as the result of the evaluation of the static criteria are passed as input to ***Percolator***. The reporter's confidence in the source is static information since the reporter encodes it in the text in the form of the lexical realisation of the source. The rating obtained from the evaluation by the text analysis component is provided to ***Percolator*** as a parameter to the **Belief** constructor method.

The second criterion, the reporter's confidence in the source is also obtained from the text analysis component. It is static information that is encoded in the text through the reporter's choice of reporting verb. The rating obtained from the evaluation by the text analysis component is provided to *Percolator* as a parameter to the **Belief** constructor method.

The third criterion, the reader's confidence in the source, is not static information since it is not found in the text. The reader's confidence in the source is evaluated each time the reader is exposed to the belief as part of the percolation of the belief to the reader's topic environment. The algorithm for the evaluation of the reader's confidence in the source is as follows:

```
for each source name in the sourceNameList
    - get the source name
    - see if the reader has any beliefs about the source

      for each belief about source name
          - evaluate each belief (if any) to determine whether
            it reflects HIGH, NEUTRAL or LOW confidence in the source
          - if it evaluates to HIGH
               increment the HIGH accumulator
            else if it evaluates to LOW
               increment the LOW accumulator
            else
               increment the NEUTRAL accumulator

 if HIGH accumulator is greatest
    return a rating of HIGH
 else if LOW accumulator is greatest
    return a rating of LOW
 else
    return a rating of NEUTRAL
```

For example, if the reader has two beliefs about any of the sources that evaluate to a HIGH rating and one belief that evaluates to a LOW rating, the HIGH accumulator will have the greatest value so a rating of HIGH will be returned.

This evaluation is done each time a belief is percolated because the sources for each exposure may be different. The rating is then stored as a data member of the **SubSource** for this exposure and added to the **sourceList** of the **Belief** object.

The fourth criterion, the reader's confidence in the reporter, is also not static information and is not encoded in the text. Like the reader's confidence in the source, it is entirely dependent upon the current nesting of environments. It is evaluated each time the reader is exposed to the belief as part of the percolation mechanism. The algorithm for the evaluation of the reader's confidence in the reporter is as follows:

```
get the reporter's name
see if the reader has any beliefs about the reporter
```

```
for each belief about the reporter
    - evaluate the belief to determine whether it reflects
      HIGH, NEUTRAL or LOW confidence in the reporter
    - if it evaluates to HIGH
         increment the HIGH accumulator
      else if it evaluates to LOW
         increment the LOW accumulator
      else
         increment the NEUTRAL accumulator

if HIGH accumulator is greatest
   return a rating of HIGH
else if LOW accumulator is greatest
   return a rating of LOW
else
   return a rating of NEUTRAL
```

The rating is stored as a data member of the **SubSource** representing the current exposure to the belief. It is evaluated each time the belief is percolated because for each exposure, the reporter may be different.

```
System believes
  Reader believes the following:

          HELD Beliefs about President
            ~sexual_relations(president,intern)
            democrat(president)

          HELD Beliefs about P. Jenings
            good(P. Jenings)
            experienced(P. Jenings)

          HELD Beliefs about S. Peters
            good(S. Peters)
            ethical(S. Peters)

          HELD Beliefs about Intern
            bad(intern)
            inexperienced(intern)
```

Figure 30: Viewpoint of a Reader with Prior Beliefs

Figure 30 shows a reader who has some prior beliefs about the topic, *President*, about the reporter, *P. Jenings*, and about the sources, *S. Peters* and *Intern*. Figure 31 shows how these beliefs were evaluated during percolation of the belief sexual_relations(president,intern) to provide ratings for the reader's confidence in the source and the reader's confidence in the reporter.

As expected, the reader's confidence in the reporter is HIGH since the reader has two prior beliefs about *P. Jenings* both of which evaluate to HIGH. The reader's confidence in the sources evaluates to NEUTRAL since there are two beliefs that evaluate to HIGH and two beliefs that evaluate to LOW.

```
System believes
  Reader believes the following:

          HELD Beliefs about President
            ~sexual_relations(president,intern)
            democrat(president)

          HELD Beliefs about P. Jenings
            good(P. Jenings)
            experienced(P. Jenings)

          HELD Beliefs about S. Peters
            good(S. Peters)
            ethical(S. Peters)

          HELD Beliefs about Intern
            bad(intern)
            inexperienced(intern)

          POTENTIAL Beliefs about President
            sexual_relations(president,intern)
              Source 1: Intern
              Source 2: S. Peters
              Reporter: P. Jenings
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      HIGH
```

Figure 31: Viewpoint of a Reader with Prior Beliefs after Percolation

7.5.2 A Formula for Evaluating the Criteria

Whenever the promotion mechanism is activated, the criteria ratings need to be evaluated to determine whether or not the reader has enough confidence in the belief's reliability to promote it to a held belief. Two important relationships among the four criteria need to be maintained in any formula used to evaluate the criteria ratings.

First, the reporter's confidence in the source and the reporter's confidence in the reported information are related to the reader's confidence in the reporter. They are only important to the extent that the reader has confidence in the reporter. For instance, if the reader has high confidence in the reporter, then the fact that the reporter has high confidence in the source and in the reported information is of more importance. If the reader has low confidence in the reporter, than his level of confidence in the source and in the reported information is of lesser importance.

Second, the reporter's confidence in the source is related to the reader's confidence in the source. In fact, the reader's confidence in the source must always have more importance than the reporter's confidence in the source regardless of the reader's confidence in the reporter. For instance, if the reader has low confidence in the source, as indicated by a prior belief about the source, his confidence in the source will not increase because the reporter has high confidence in the source.

Formula (7), used to evaluate the criteria in *Percolator*, reflects both these relationships.

(7) $Reliability\ of\ the\ belief = (S1 + (R * I) + S1(R * S2))$
where
S1 is the reader's confidence in the source,
R is the reader's confidence in the reporter,
S2 is the reporter's confidence in the source, and
I is the reporter's confidence in the reported information
S1, R, S2, I can have a value of HIGH = 1.5, NEUTRAL = 1 or LOW = 0.5.

The first relationship, that between the reporter's confidence in the source and in the reported information and the reader's confidence in the source, is reflected in the formula by multiplying I by R and S2 by R. If the reader's confidence in the reporter is HIGH, then the reporter's confidence in the source and in the reported information will have greater importance (or weight in the formula) since each will be multiplied by 1.5. If the reader's confidence in the reporter is LOW, then the reporter's confidence in the source and the reporter's confidence in the reported information will have less importance since they will be multiplied by 0.5. On the other hand, if the reader's confidence in the reporter is NEUTRAL, then the reporter's confidence in the source and in the reported information will be taken at face value since they will be multiplied by 1.

The second relationship, that between the reader's confidence in the source and the reporter's confidence in the source, is reflected in the formula by multiplying the S2 (or rather (R * S2) as indicated by the first relationship) by S1. If the reader has LOW confidence in the source then the reporter's confidence in the source will be lowered by multiplying it by 0.5. If the reader has HIGH confidence in the source then the reporter's confidence in the source will be increased by multiplying it by 1.5. If the reader's confidence in the source is NEUTRAL, then the reporter's confidence in the source is taken at face value since it will be multiplied by 1.

This method of implementing the reader's confidence in the source enforces the fact that this is the most important criterion. For instance, if John has low confidence in Jack, it really doesn't matter what Mary says Jack said.

Taking the potential belief shown in Figure 31 , the reliability of the belief would be evaluated as follows for that particular exposure:

Let

S1, the reader's confidence in the source, = NEUTRAL = 1
R, the reader's confidence in the reporter, = HIGH = 1.5
S2, the reporter's confidence in the source, = HIGH = 1.5
I, the reporter's confidence in the reported information, = HIGH = 1.5

(8) $\textit{Reliability of the belief} = 5.5 = (1 + (1.5 * 1.5) + 1(1.5 * 1.5))$

The maximum reliability that can be evaluated by this formula is 7.125 if all values are HIGH. The lowest reliability that can be evaluated is 0.875 if all values are LOW. If all values are NEUTRAL then the reliability will be 3.0. The next step in the calculation is to convert the result reliability back to a rating of HIGH, NEUTRAL or LOW. This is done by taking the range 7.125 - 0.875 = 6.25 and dividing it into three sub-ranges. Reliability ratings that fall within the range 0.875 to 2.954 will be converted to a LOW rating. Reliability ratings within the range 2.955 to 5.034 will be converted to a NEUTRAL rating and reliability ratings within the range 5.035 to 7.125 will be converted to a HIGH rating. This rating is the value that is returned to the *belief heuristic*.

The reliability of the belief in Figure 31 for that particular exposure evaluates to 5.5. So, a reliability rating of HIGH will be returned to the belief heuristic.

7.5.3 Belief Heuristic

A *belief heuristic* is the manner in which the reporter's confidence in the source, the reporter's confidence in the reported information, the reader's confidence in the reporter, the reader's confidence in the source as well as any previous beliefs held by the reader are combined to form a comprehensive quantitative evaluation of the reliability of a belief. It is the belief heuristic that will determine whether or not a potential belief can be promoted to a held belief. Also, as was discussed in Section 6.4.6, the belief heuristic must be able to take into consideration the fact that repeated exposure to a belief will often eventually lead to the acceptance of the belief.

The belief heuristic used in *Percolator* incorporates all of these factors. As was discussed in Section 7.5.1, the reporter's confidence in the source and the reporter's confidence in the reported information are obtained from the reporter's encodings in the text of the news article. The reader's confidence in the source(s) and the reader's confidence in the reporter are obtained by evaluating the reader's beliefs (if any) about the source(s) and the reporter for each exposure to the belief. These ratings are then combined using the formula in Definition 7. This formula returns a rating of HIGH, NEUTRAL or LOW reliability for a particular exposure to a belief. The belief heuristic will ensure that this evaluation is done for each exposure to a potential belief using the ratings as stored in the **SubSource** object representing each exposure.

At this point, the belief heuristic needs to combine the evaluation of each exposure to calculate the overall reliability of the belief. It is this overall reliability that will determine whether or not the potential belief can be promoted to a held belief. In order to do this, the belief heuristic will assign a value to the rating of each exposure to the belief. If the rating of the exposure is HIGH as calculated by the formula in Definition 7, a value of 0.75 is assigned. If the rating is LOW, a value of 0.25 is assigned and if the rating is NEUTRAL, a value of 0.5 is assigned. The values of each exposure are summed up and if the result is larger than the threshold, the potential belief can be promoted to a held belief.

Percolator uses two thresholds. The first threshold, which has a value of 1.5, is used if the reader has no prior opposing beliefs about the topic. It is assumed that it is easier for a reader to accept a belief as a held belief if he has no prior opposing beliefs about the topic. The threshold of 1.5 can be achieved using any combination of HIGH, LOW, or NEUTRAL exposures. For example, a reader would need to be have two exposures rated HIGH to reach the threshold (0.75 + 0.75). The reader could also have three exposures rated NEUTRAL (0.5 * 3). Even if each exposure to the belief that the reader has is rated LOW, the reader will eventually accept the belief. It will be promoted to a held belief after six exposures rated LOW. After repeated exposures to a belief, even low reliability exposures will eventually lead to acceptance of the belief.

The second threshold, which has a value of 2.25, is used if the reader has opposing beliefs about the topic. It is assumed that it is more difficult for a reader to accept a belief as a held belief if he has opposing beliefs since doing so would reverse a held belief. For instance, it would take three HIGH rated exposures for the potential belief to be promoted to a held belief instead of the two that are needed for a reader with no prior opposing beliefs about the topic. An opposing belief can be (1) the negation of a belief where P is the belief and $\neg P$ is the opposing belief or (2) the opposite of a belief such as dry(towel) when the belief is wet(towel). All systems should be able to handle opposing beliefs of type one. However, the large lexica and sophisticated reasoning necessary to support opposing beliefs of type two are beyond the scope of this thesis. Therefore, *Percolator* only handles opposing beliefs of type one.

The algorithm for the belief heuristic used in *Percolator* is the following:

```
for each Potential belief in the Reader's topic environment
    - verify if the reader has an opposing belief
    - if there are opposing beliefs
        the threshold used is 2.25
      else
        the threshold used is 1.5
    - get the source list of the current belief

    for each subsource in the source list
```

```
      - evaluate the ratings (using the formula)
      - if the result is LOW
          add 0.25 to the overall reliability
        else if the result is NEUTRAL
          add 0.5 to the overall reliability
        else if the result is HIGH
          add 0.75 to the overall reliability

  - if overall reliability >= threshold
        promote the belief
```

At this point, the promoting of the belief simply consists of adding the belief to the list of held beliefs (**heldBel**) in the reader's topic environment and removing it from the list of potential beliefs (**potentialBel**). In addition, the source list is removed since the source list is no longer needed. This was a design decision based on the fact that source lists can not be kept indefinitely and on the assumption that at some time reader's generally forget or no longer need to remember the source of beliefs that they have accepted as held beliefs. It was decided that the time to remove the source list would be when the belief is promoted to a held belief.

```
System believes
  Mary believes the following:

          POTENTIAL Beliefs about Common Cold
            cure(common_cold,Dr. J. Smith)
              Source 1: Dr. J. Smith
              Reporter: M. Johnson
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

              Source 1: Dr. R. Goldberg
              Reporter: S. Robinson
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

              Source 1: Dr. S. Donaldson
              Reporter: K. Johanson
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL
```

Figure 32: State of Mary's Viewpoint After Three Exposures to a Belief

As an example, consider the case presented in Figure 27, Mary, the reader who is exposed to the belief that Dr. J. Smith has found the cure for the common cold by the reporter M. Johnson. Mary has no prior beliefs about the reporter or the source. She also has no prior beliefs about the topic, common cold. Figure 32 shows the state of Mary's viewpoint after three exposures to the belief cure(common_cold,Dr. J. Smith). For each exposure to the belief, the rating of the four confidence criteria is shown as well as the names of the source(s) and reporter.

```
System believes
  Mary believes the following:

          HELD Beliefs about Common Cold
            cure(common_cold,Dr. J. Smith)
```

Figure 33: State of the Mary's Viewpoint After Promotion of a Belief

Figure 33 shows the state of Mary's viewpoint after the belief has been promoted to a held belief. In this case, it took three exposures to the belief in order to have an overall belief reliability rating that could achieve the threshold. In this case, the threshold is set to 1.5 since Mary has no prior opposing beliefs. As can be seen in the figure, the belief has been removed as a potential belief and added as a held belief. In addition, the source list has been removed since it is no longer needed.

Chapter 8

Experiments with *Percolator*

This chapter will show how the theories and mechanisms presented in this thesis can be used in concert to simulate how a reader of news articles would acquire beliefs. It will present a sample news article recently taken from the Montreal Gazette web site. First, this chapter will illustrate the procedure outlined in [Bergler, 1992, Bergler, 1995a] to extract profile structures, the reporting verbs, and the source lexicalisations from the article and evaluate these components. Then the viewpoint nestings will be created in *Percolator* and the beliefs extracted from the profiles will be passed into the appropriate viewpoints.

Three different types of user models will be created: that of a reader with no previous beliefs about the topic, a reader with supporting beliefs about the topic and a reader with opposing beliefs about the topic. *Percolator* will be used to percolate the beliefs taken from the news article into the different user models to show how this article impacts differently on each of the models.

It should be noted that *Percolator* was designed as a stand-alone mock-up of a single component of a large and complex system. A complete system would at least include a reasoner, belief maintenance, belief revision, belief ascription, text analysis (which incorporates evidential analysis and profiling) and a semantic dictionary. Since *Percolator* is a stand-alone proof-of-concept system several strongly simplifying assumptions were made. However, each assumption has some justification relative to the current application. The following lists the assumptions and the reasoning behind them.

1. belief percolation and promotion apply to only one belief at a time.

   All propositions about a particular topic by one source will be processed one after another. Contradictions can exist in one article when an article contains opposing supporting groups. By processing the beliefs about a particular topic from one source at a time, the contradictions that arise will occur within a particular topic environment and are dealt with one at a time by the promotion mechanism.

2. a structure specific to modelling a reader of news articles is assumed.

   The consequence of this assumption is that while the reader can have several topic boxes in his viewpoint, the other agents in the nesting are restricted to one. The goal of the system is to build up a model of the reader. This means that the reader's viewpoint will eventually contain a "belief base" of sorts containing beliefs about all the information acquired from reading news articles. This is analogous to the use of the system viewpoint in ViewGen, Ballim's implementation of his ViewFinder framework. However, the viewpoint of the other agents in the nesting are temporary and since only beliefs about one topic are percolated and promoted at a time, these agents only require one topic environment. The reporter may have beliefs about the source which the system acquires through evidential analysis. In Percolator, a decision was made to maintain this information in the source list. As such, the reporter's viewpoint, also, only requires one topic environment.

3. a particular form of evidential analysis is assumed.

   Evidential analysis is only performed on the reporting verb and the lexicalisation of the source and a single-valued result is supplied to *Percolator*. Evidential analysis is dependent on several factors, including the user and the purpose of the search. A search can conceivably apply two different evidential analyses to the same text (to cover all possibilities). The fact that *Percolator* evaluates the results of the evidential analysis in a method, allows for other forms of evidential analysis to be added to the system at a later date.

These assumptions limit the generality of the system but for good reasons. Thus, while the principles behind *Percolator* are based on Ballim's framework, *Percolator* is a particular application of the principles with the required adjustments.

8.1 Profiling a Sample News Article

In order to demonstrate how *Percolator* and the theories presented in this thesis could be used to construct models of readers of news articles, it is necessary to select one of many typical applications. The application that was selected is one of a search engine that incorporates an agent that would scan the Internet for news articles that might interest the user. The agent would use a system like *Percolator* to construct a model of the user's beliefs to assist the agent in determining which articles might be relevant for the user. *Percolator* would allow the agent to build up the model of the user's beliefs based on the content of the articles presented to the user and that the user indicates as being useful.

The sample news article, shown in Figure 34, is the type of article that might be picked up by a search engine with the keywords "abortion" or "hate group". It was obtained from the Internet web site of the Montreal Gazette on July 12, 2000. It follows the typical North American reporting style. That is to say, the author quotes a source on all controversial statements. He also provides quotes from people with differing points of view concerning the topic, those who believe in the link between anti-abortion groups and hate groups and those who don't. Only the more applicable parts of the article have been retained for the example. The lexical realisation of the source which encodes the reporter's confidence in the source is indicated in *italic* print. The reporting verb which encodes the reporter's confidence in the reported material is indicated in **bold** print.

Anti-abortion Radicals Linked to Hate Groups, Says Author

STEVE MERTL

VANCOUVER (CP) - Anti-abortion extremists have long-standing ties to the radical right wing, **says** *a lawyer and author of two books on hate groups.*

"The connection's been there for anybody who takes a look at the far right," *Warren Kinsella* **said** Wednesday in an interview from Toronto. *Intelligence experts* **believe** those responsible for attacks on abortion doctors have had help eluding police, *Kinsella* **said**. *A spokeswoman for an anti-abortion group* **hotly denied** that anyone in her mainstream movement would advocate such behaviour, and *Kinsella* **agreed**.

"A lot of them run in the opposite direction from these guys," *he* **said**. "But there are ones who play footsie with the far right."

...

Officials with the Winnipeg-based police task force investigating the abortion-related attacks, as well as the *Canadian Security Intelligence Service*, **would not** say whether they are probing connections between the anti-abortion movement and hate groups. "We investigate politically motivated violence," *CSIS spokesman Dan Lambert* **said** from Ottawa.

"It's been pretty clear in the past that we are involved in investigations of right-wing individuals who profess to be involved in serious violence."

Abortion has been on the far-right gripe list for decades, **said** *Kinsella, whose 1994 book Web of Hate looked at Canadian hate groups.*

...

"Two themes have been really seized upon to boost membership and attract people - homophobic kind of messages and particularly, anti-abortion stuff," *he* **said**.

"All of them, from the Heritage Front to the Western Canada Concept, go on and on and on about the perils posed by abortion and how it is a threat to the white race and Christian principles."

...

Kinsella **said** anti-abortion groups are fertile recruiting sites for hate groups that lure young people with a certain profile - white, twentyish, single and socially outcast.

But *Monica Roddis, of the B.C. Pro-Life Society*, **firmly denied** groups such as hers harbour violent extremists.

"In all of that time in all our meetings, we have never encouraged, we've never even had people present violence as an option," **said** *Roddis*.

"We don't have people coming into our meetings and saying, 'I'm thinking of bombing a clinic.' "

Roddis **said** she'd report anyone who suggested attacking doctors to the police.

...

But that doesn't mean the group will temper its message that abortion is murder, *she* **said**.

However, *Kinsella* **maintains** the links to extremist are there.

The so-called Nuremberg files are the best example of that cross-over, *he* **said**.

...

Figure 34: Montreal Gazette, July 12, 2000

A profile is a collection of information pertaining to one specific source in a news article. An entire text can be represented in a complex profile structure. Bergler [Bergler, 1992] points out several heuristics that can be used to build the profile structure of an article. These are described in Section 3.5 and will be used here to construct the complex profile structure of the article in Figure 34.

The first sentence in an article often consists of a summary of the main points of the article. In addition, the principal or root source is usually the first source introduced in the article. The root source is usually the principal source for the information supporting the reporter's intent. In our sample text, the first sentence is *Anti-abortion extremists have long-standing ties to the radical right wing, says a lawyer and author of two books on hate groups.* This indicates that the article is about the connections between anti-abortion extremists and the radical right wing. It introduces the principal source, Warren Kinsella (named later in the article). Mr. Kinsella supports the notion that there is a connection between the two groups. Since Mr. Kinsella is the root source, this would indicate that the reporter, Steve Mertl intends to support this opinion also. As concerns the construction of the complex profile structure, all the other sources in the article will either be confirming or contradicting the root source.

The second heuristic contends that the first sentence of the text usually mentions a conflict of interest if there is one. In our example, this would indicate that there are (at least) two groups of people: those who think that some members of anti-abortion groups are linked to the radical right and those who think that there is no link. All the sources that agree with the root source will comprise one *supporting group*. All the sources that contradict the root source will also form one *supporting group*. And, since there is a conflict of opinion in this article, the groups will form *opposing supporting groups*.

Believe in Link between anti-abortion extremists and radical right wing

Warren Kinsella, a lawyer and author of two books on hate crimes, whose 1994 book Web of Hate looked at Canadian hate groups

- says anti-abortion extremists have long standing ties to the radical right wing
- said "The connection's been there for anybody who takes a look at the far right."
- said Intelligence Experts believe those responsible for attacks on abortion doctors have had help eluding police.
- agreed that [people] in a mainstream movement would not advocate such behaviour.
- said "A lot of them run in the opposite direction from these guys."
- said "there are ones who play footsie with the far right."
- said abortion has been on the far-right gripe list for decades.
- said two themes have been really seized upon to boost membership and attract people - homophobic kind of messages and particularly, anti-abortion stuff.
- said, "All of them, from the Heritage Front to the Western Canada Concept, go on and on and on about the perils posed by abortion and how it is a threat to the white race and Christian principles.
- said anti-abortion groups are fertile recruiting sites for hate groups that lure young people with a certain profile - white, twentyish, single and socially outcast.
- said the links to extremists are there.
- said the Nuremberg files are the best example of that cross-over.

Officials with the Winnipeg-based police task force investigating the abortion-related attacks

- would not say whether they are probing connections between anti-abortion movement and hate groups.

Canadian Security Intelligence Service

- would not say whether they are probing connections between anti-abortion movement and hate groups.

CSIS spokesman, Dan Lambert

- said "We investigate politically motivated violence."
- said "It's pretty clear in the past that we are involved in investigations of right wing individuals who profess to be involved in serious violence."

Don't Believe in Link between anti-abortion extremists and radical right wing

A spokeswoman for an anti-abortion group

- hotly denied that anyone in her mainstream movement would advocate such behaviour

Monica Roddis, of the B.C. Pro-Life Society

- firmly denied groups such as hers would harbour violent extremists
- said "In all of that time in our meetings, we have never encouraged, we've never even had people present violence as an option."
- said "We don't have people coming into our meetings and saying, 'I'm thinking of bombing a clinic.' "
- said she would report anyone who suggested attacking doctors to the police.
- said that doesn't mean the group will temper down its message that abortion is murder.

The resultant complex profile structure for this article is composed of two supporting groups. The first group consists of the sources that believe in the link between anti-abortion extremists and the radical right wing. The root source is Warren Kinsella and the supporting sources are Officials with the Winnipeg-based police task force, the Canadian Security Intelligence Service (CSIS), and Dan Lambert, a spokesman for the CSIS. The other supporting group consists of the sources that don't believe in the link between anti-abortion extremists and the radical right wing. The supporting sources for this group are a spokeswoman for an anti-abortion group and Monica Roddis of the B.C. Pro-Life Society[1]. These two supporting groups are in fact opposing supporting groups.

The complex profile structure stores several pieces of information that are required by *Percolator*. The first is the reporting verb used to report the reported information. The second is the lexical realisation of the source. The third is the reported information which will be converted into a logical proposition. It also provides some idea about the relationships between the sources or agents in the article.

8.2 Evaluation of the Reporter's Encodings

Section 3.2 describes how the evaluation of the reliability of reported speech is conventionalised in *newspaper style* and that the evaluative environment is provided largely by the lexical semantics of the two functional parts of the matrix clause of the reported speech: the source and the reporting verb. Evidential analysis of the reported speech as discussed in Chapter 3 consists mainly of extracting the information encoded by the reporter in the reporting verb and in the lexical realisation of the source to determine the reporter's confidence in the reported information and the reporter's confidence in the source. The following sections will show how the reporting verb and the lexical realisation of the source are evaluated.

[1]It should be noted that a sophisticated profiler might be able to resolve these two sources as being one and the same. For this example, I chose to illustrate the case where the profiler could not resolve the identity of the two agents in order to show that the profiling of the article would work nonetheless.

8.2.1 Evidential Analysis of the Reporting Verb

The complex profile structure of the article in Figure 34 shows that for all the sources in the supporting group containing the root source, the reporter used the reporting verb ***to say*** and on one occasion he used ***to agree***. Analysis of the semantic dimensions of the reporting verb ***to say*** as described in Section 3.4 results in the determination that the reporter's confidence in the reported material is neutral. All the beliefs stemming from this supporting group, when passed to *Percolator*, will also include a rating of NEUTRAL for the reporter's confidence in the reported information.

When quoting members of the supporting group for those who don't believe in the link between anti-abortion extremists and the radical right, the reporter used the reporting verb ***to say*** as well as the reporting verb ***to deny***. Analysis of the semantic dimensions of the reporting verb ***to deny*** results in the determination that the reporter's confidence in the reported material is low[2]. The beliefs stemming from this supporting group when passed to *Percolator* will include a rating of LOW for the reporter's confidence in the reported information if the reporting verb is ***to deny*** and NEUTRAL if the reporting verb is ***to say***.

8.2.2 Evidential Analysis of the Source

The complex profile structure of the article in Figure 34 indicates the source of each of the propositions. At the top of each box is the lexical realisation of the source used by the reporter to encode his confidence in the source. Evidential analysis of the source as described in Section 8.2.2 must be conducted for each source in the news article and passed to *Percolator* along with the appropriate proposition.

The root source is a member of the first supporting group, those who believe in the link between anti-abortion extremists and the radical right wing. He is identified as ***Warren Kinsella, a lawyer and author of two books on hate crimes***. Later, it is mentioned that one of his books written in 1994, Web of Hate, looked at Canadian hate groups. Evidential analysis of this lexical realisation indicates the reporter's opinion that the source is inherently trustworthy since he is an expert on the subject of the reported speech, hate groups. The fact that he is a lawyer indicates that he is educated and perhaps has defended or been a prosecutor (or has known someone who has) on hate group cases, although this is not stated in the text. Certainly, he would have better access to such cases than non-lawyers. By addressing him as the author of two books on hate crimes, the reporter is establishing the source's expertise. Later, mentioning that one of the books was on Canadian hate crimes even more firmly establishes his expertise since several of the attacks on abortion doctors took place in Canada (including the one that prompted this article). Evidential analysis of Warren Kinsella as a source on the link between anti-abortion extremists and the radical right wing would yield a rating of HIGH for the reporter's confidence in the source.

The next source in the first supporting group is identified as ***officials with the Winnipeg-based***

---

[2] ***to deny*** is defined as "to declare something as untrue". Bergler (2000, personal communication) says that "If you declare that something is not true, that something has been suggested to you as held belief of some people. Your qualification to declare untruth stems from the fact that you are involved somehow, which immediately lowers your credibility (as a mirror image to credibility going up when you concede something). The negative polarity of course also plays a role, similar to insist.

police task force investigating the abortion-related attacks. In this case, the use of "officials" suggests some authority on the matter. Furthermore, specifying that they are part of the task force investigating the abortion-related attacks gives evidence as to the source's involvement with the subject. Evidential analysis of this source would also lead to a rating of HIGH for the reporter's confidence in the source.

The third source in the first supporting group is identified as the *Canadian Security Intelligence Service*. This is a government agency that often deals with politically motivated crimes, especially violent crimes. The author is establishing the authority of this source on the subject of hate groups. Evidential analysis would yield a HIGH rating for the reporter's confidence in this source.

The fourth and final source in the first supporting group is CSIS spokesman, Dan Lambert. The reporter establishes Mr. Lambert's authority on this matter by first, identifying him by name, then as spokesman for the CSIS. As spokesman, he has the authority to speak for the Service. And, as M. Lambert mentions, the CSIS often investigates this type of crime and in the past has investigated right wing individuals involved in serious violence. Evidential analysis of this source would yield a HIGH rating for the reporter's confidence in this source.

The second supporting group consists of two sources. The first is identified as *a spokeswoman for an anti-abortion group*. The reporter gives her some authority by calling her a spokeswoman for the group but does not identify her by name nor does he identify for which anti-abortion group she speaks. Evidential analysis of this source would likely indicate LOW or NEUTRAL confidence since the source is anonymous. For the purposes of this example, a rating of LOW will be given for the reporter's confidence in this source.

The second source in the second supporting group is identified as *Monica Roddis, of the B.C. Pro-Life Society*. The reporter establishes her involvement by indicating that she is a member of an anti-abortion group which he identifies by name. He also identifies Ms. Roddis by name which establishes some measure of trustworthiness. Evidential analysis of this source would yield a rating of HIGH for the reporter's confidence in this source.

These ratings determined for the reporter's confidence in the source and the reporter's confidence in the reported information will be passed along to *Percolator* with the belief after the viewpoints and appropriate topic environments have been created.

8.3 Creating the Viewpoints

When the tasks of creating the complex profile structure of the news article and conducting an evidential analysis of the reporting verb and the source have been completed, the viewpoints can be created with the appropriate nesting. Again, the complex profile structure can be of some use. In order to create the viewpoints, it is necessary to identify all the agents that come into play for each proposition. The complex profile structure is a good starting point for this since it identifies the sources of each statement. However, it is important to look within the statement for other sources. Once the agents have been identified, the viewpoints can be created.

The outer viewpoint in any nesting in *Percolator* is always the *system* viewpoint. Since the purpose of *Percolator* is to model the reader's beliefs, the system must view the article from the reader's viewpoint. This means that the reader's viewpoint must be *within* the system's viewpoint. Given that the purpose is also to model the beliefs of readers of news articles and that news articles are written by a reporter, we must include the reporter in the nesting. By virtue of the article having been written by the reporter, the reader will view the events in the article from the reporter's viewpoint. Thus, the reporter's viewpoint will be *within* that of the reader. In accordance with typical North American news reporting style, the reporter gets the reported information from a source which he cites in the article. The reporter is also viewing the reported information through the source. Again, this is represented by placing the source's viewpoint within that of the reporter. If in turn, the source is quoting another source, then the second source's viewpoint is placed within that of the first source, and so on. The nesting of viewpoints just described is shown in Figure 35. It is the standard pattern for most situations.

The system, reader, reporter and source are essential roles in the modelling of the reader's beliefs and will always be present as shown in Figure 35. They are called roles since in an actual nesting they are assigned to an actual agent. In the example, the system is *Percolator* (but the name *system* will still be used for this viewpoint), the reader is the user of the system (or whichever reader model is being applied), the reporter is Steve Mertl, and the source is one of the sources in the two supporting groups depending on the proposition. However, as previously mentioned, the source from one of the supporting groups can be citing another source not in the supporting groups. One example of this from the sample text is the statement by Kinsella that *Intelligence experts believe those responsible for attacks on abortion doctors have had help eluding police.* In this case, *Intelligence experts* would be the original source.

After the viewpoints have been created, the topic environment of the source can be created. The topic environment depends on the subject of the belief in question. The granularity of the topic environment depends on the needs of the application. In some applications, it might be necessary to have a very fine granularity. In other applications, general groupings might be enough. Beliefs stemming from the sample text might fall under several topics such as abortion, anti-abortion extremists, anti-abortion group, anti-abortion radicals, hate group, right wing radicals, etc., for a fine granularity. Or, anything related to abortion could fall under one big topic. Another could be used for the right wing radicals and hate group related statements. Having a coarse granularity might imply the need for more logic in the system in order to have the ability to associate the topics. For example, when faced with a belief about pro-life groups, the system would need to know to look under the topic *abortion* for possible supporting or opposing beliefs.

The first statement of the root source will be used as an example. The source viewpoint will be made for Warren Kinsella. A topic environment about anti-abortion will be created in the source's viewpoint. If the reader (or reader model being used) has beliefs about anti-abortion, then a topic environment will be created for the reader too. The idea here is that there needs to be a topic environment in which to put the beliefs. No topic environment needs to be created when there are no previous beliefs, as in the case of the reporter, since one will be created by the system during percolation.

The belief ties(anti-abortion_extremists,radical_right_wing) will be added to the source's viewpoint along with rating from the evaluation of the reporter's confidence in the source and the reporter's confidence in the reported information. As previously mentioned, these ratings will only be printed out for the reader viewpoint. In this example, the rating for the reporter's confidence in the reported information is NEUTRAL and the rating for the reporter's confidence in the source is HIGH. Initially, the state of the system is as shown in Figure 35.

```
System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
            play_footsie(anti-abortion_extremists,far_right)
            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 35: Initial State of System

8.4 Creating Reader Models

A reader can have beliefs on many subjects prior to reading about the subject. These beliefs can be in accordance or discordance with the beliefs reported in the newspaper articles. For the purposes of this example, the models for three types of reader will be created. In the first type, the reader is a naive reader. This is a reader who has no previous beliefs about the topic. The reader in Figure 35 is an example of this type of reader.

The second model is that of a reader with one or more supporting beliefs. At this time in its evolution, *Percolator* considers a belief to be a supporting belief of the new belief if the propositions of both are an exact match. Furthermore, a held belief can not act as a supporting belief for the new proposition. Since the propositions are an exact match, it does not make much sense to percolate to the reader a belief that they already have as a held belief.

The more interesting situation is that of the reader who has a potential belief that supports the new belief. This means that the reader has already been exposed to the belief. In this situation, the source list for the potential belief is updated with the information from the new belief. After several exposures, the potential belief will be promoted to a held belief according to the belief heuristic as discussed in Section 7.5.3. Figure 36 shows the initial state of the system for a reader with a supporting belief.

The reader has been given the potential belief ties(anti-abortion_extremists,radical_right_wing). It's source list shows that, for the first exposure to the belief, the source was James Forester and the

```
System believes
  Reader believes the following:

        POTENTIAL Beliefs about anti-abortion
          ties(anti-abortion_extremists,radical_right_wing)
            Source 1: James Forester
            Reporter: Maggie Smith
              Reporter's confidence in the Source(s):     HIGH
              Reporter's confidence in the Information: HIGH
              Reader's confidence in the Source(s):       NEUTRAL
              Reader's confidence in the Reporter:        NEUTRAL


System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

        Beliefs about anti-abortion
          ties(anti-abortion_extremists,radical_right_wing)
          run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                         far_right)
          play_footsie(anti-abortion_extremists,far_right)
          use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
          fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 36: Initial State of System for Reader Model with a Supporting Belief

reporter was Maggie Smith. The rating for the reporter's confidence in the source is HIGH. The rating is also HIGH for the reporter's confidence in the information. The ratings for the reader's confidence in the source and in the reporter are both NEUTRAL.

The third model is that of a reader who has an opposing belief to the new belief. A reader can have an opposing held belief or an opposing potential belief. If the reader has an opposing held belief, the new belief will still be percolated into the reader's topic environment as a potential belief. However, the consequence of the reader having an opposing held belief is that the value of the reliability threshold determining when a potential belief will be promoted to a held belief will be higher. This type of reader is harder to convince since it involves the reversal of a previously held belief. A reader can also have an opposing potential belief. Such a belief is not considered by *Percolator* when percolating or promoting a belief. Both beliefs are kept as potential beliefs. Having opposing potential beliefs does not cause a contradiction for *Percolator* since only held beliefs are considered during promotion. Eventually, the reader will have sufficient confidence in one of the contradicting beliefs for it to be promoted to a held belief. The other belief will, nevertheless, be kept as a potential belief.

In order to make it a more interesting model, the reader has one opposing held belief and one opposing potential belief with an accurate source list. Figure 37 shows the initial state of

```
System believes
  Reader believes the following:

        HELD Beliefs about anti-abortion
          ~ties(anti-abortion_extremists,radical_right_wing)

        POTENTIAL Beliefs about anti-abortion
          ~fertile_recruiting_sites(anti-abortion_groups,hate_groups)
            Source 1: President Heritage Front
            Reporter: Maggie Smith
              Reporter's confidence in the Source(s):    NEUTRAL
              Reporter's confidence in the Information: LOW
              Reader's confidence in the Source(s):      NEUTRAL
              Reader's confidence in the Reporter:       NEUTRAL


System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

        Beliefs about anti-abortion
          ties(anti-abortion_extremists,radical_right_wing)
          run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                          far_right)
          play_footsie(anti-abortion_extremists,far_right)
          use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
          fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 37: Initial State of System for Reader Model with Opposing Beliefs

the system for a reader with opposing beliefs. The reader has been given the held belief ties(anti-abortion_extremists,radical_right_wing) and the potential belief fertile_recruiting_sites(anti-abortion_groups, hate_groups). To keep the models simpler, the reader has no prior beliefs about the reporters and the sources.

8.5 Percolating the Beliefs

Belief percolation as a method of simulating how a reader of news articles acquires beliefs from the articles was first introduced in Chapter 6. Section 7.3 explained how belief percolation is implemented in *Percolator*. This section will look at how the three reader models defined in Section 8.4 are impacted upon by the percolation mechanism. The beliefs in the source's topic environment, as shown in Figures 35, 36, and 37 depicting the initial state of the system for each of the three models, will be percolated through the nesting of viewpoints. The following sections will look at three cases as defined by the reader models. Case one will show how *Percolator* simulates how a naive reader (model 1) acquires beliefs by reading the sample text in Figure 34. The second case will show how *Percolator* simulates this acquisition process by a reader with previous supporting beliefs (model 2). Finally, the third case will show how *Percolator* simulates this acquisition process by a reader with previous opposing beliefs (model 3).

8.5.1 Case 1: No Previous Beliefs

The reader in this case is one of the first model. The reader has no previous beliefs about the topic. Continuing the example of the previous sections, the topic in question is anti-abortion. Furthermore, the reader has no previous beliefs about the reporter or the source. (See Section 7.5.1 for a discussion on the evaluation of previous beliefs about the reporter and the source.)

The initial state of the system for this case is shown in Figure 35. The source, Warren Kinsella, has five beliefs in his anti-abortion topic environment. The reader's viewpoint does not contain an anti-abortion topic environment since the reader has no beliefs about this subject at this point.

```
System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
            play_footsie(anti-abortion_extremists,far_right)
            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 38: Source's Viewpoint after Percolation - Model 1

Figures 38, 39, and 40 show the state of the viewpoints of the source, the reporter, and the reader, respectively, after percolation of the beliefs has occurred. As expected, the source's viewpoint has not been affected by the percolation mechanism. There is no reason why the source's beliefs should be changed by the simple act of telling his beliefs to the reporter!

On the other hand, the reporter's viewpoint, shown in Figure 39 has been changed by the percolation mechanism. It now contains an anti-abortion topic environment into which the beliefs in the source's anti-abortion topic environment have been percolated. As previously stated, only the reader's topic environments will distinguish between held and potential beliefs. In addition, the source list for each belief has been started. At this point all that is known is that the source is Warren Kinsella.

The reader's viewpoint, shown in Figure 40, has also been greatly affected by the percolation mechanism. Firstly, the new topic environment, anti-abortion has been created. Secondly, all the beliefs in the reporter's anti-abortion topic environment have been percolated into the reader's anti-abortion topic environment as ***potential beliefs***. And, thirdly, the source list has been updated in

```
System believes
  Reader believes
    Steve Mertl believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
              Source 1: Warren Kinsella

            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
              Source 1: Warren Kinsella

            play_footsie(anti-abortion_extremists,far_right)
              Source 1: Warren Kinsella

            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
              Source 1: Warren Kinsella

            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
              Source 1: Warren Kinsella
```

Figure 39: Reporter's Viewpoint after Percolation - Model 1

several ways. The name of the reporter has been added as have the ratings for the reporter's confidence in the source and the reporter's confidence in the reported information. In addition, the reader's confidence in the source and the reader's confidence in the reporter have been evaluated as described in Section 7.5.1.

The next step that *Percolator* would take is to decompose the nestings since they are no longer needed. All that would remain is the system's viewpoint and the reader's viewpoint including the anti-abortion topic environment. The reader in this case has become a reader with previous beliefs about anti-abortion since he now has several potential beliefs in his anti-abortion topic environment. If this reader is re-exposed to these potential beliefs, they can, when the reader has sufficient confidence in their reliability, be promoted to held beliefs. The promotion mechanism and the belief heuristic that guides it are discussed in Section 7.5.

```
System believes
  Reader believes the following:

          POTENTIAL Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

            play_footsie(anti-abortion_extremists,far_right)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL

            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):   HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):     NEUTRAL
                Reader's confidence in the Reporter:      NEUTRAL
```

Figure 40: Reader's Viewpoint after Percolation - Model 1

8.5.2 Case 2: Supporting Beliefs

The reader in this case is of the second model. The reader has a previous supporting belief about the topic. As previously discussed, for this belief to have an impact, it must be a potential belief. Figure 36 shows the initial state of the system for this case. The reader has the previous potential belief ties(anti-abortion_extremists, radical_right_wing). The source list for this belief shows that the reader has been exposed to the belief once. It also shows the names of the source and the reporter. The rating for the reporter's confidence in the source and the reporter's confidence in the reported information are both HIGH. The reader's confidence in the source and in the reporter are both rated NEUTRAL since, in this example, the reader has no previous beliefs about the source or the reporter. In addition, the source Warren Kinsella, has five beliefs in his anti-abortion topic environment. One of the source's beliefs is the same as the reader's potential belief ties(anti-abortion_extremists, radical_right_wing).

```
System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
            play_footsie(anti-abortion_extremists,far_right)
            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 41: Source's Viewpoint after Percolation - Model 2

Figures 41, 42, and 43 show the state of the viewpoints of the source, the reporter, and the reader, respectively, after the percolation of the source's beliefs has occurred. Again, as expected, the source's viewpoint has not changed.

The reporter's viewpoint has been changed in several ways as in Case 1. It now contains an anti-abortion topic environment into which the beliefs in the source's anti-abortion topic environment have been percolated. In addition, the source list for each belief has been started. At this point all that is known is that the source is Warren Kinsella. The changes to the reporter's viewpoint are the same in this case as in the first case, that of a reader with no previous beliefs about the topic, since the contents of the reader's viewpoint have no impact on the contents of the reporter's viewpoint.

As for the reader's viewpoint, it too has changed. No new topic environment for anti-abortion has been created since one already existed. The four beliefs in the reporter's topic environment that are unsupported by beliefs in the reader's topic environment were percolated as new potential beliefs. However, the belief ties(anti-abortion_extremists, radical_right_wing) which is supported by a belief in the reader's anti-abortion topic environment was not percolated as a new potential belief. Rather, information about this new exposure to the belief was added to the belief's source list in the reader's

```
System believes
  Reader believes
    Steve Mertl believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
              Source 1: Warren Kinsella

            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
              Source 1: Warren Kinsella

            play_footsie(anti-abortion_extremists,far_right)
              Source 1: Warren Kinsella

            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
              Source 1: Warren Kinsella

            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
              Source 1: Warren Kinsella
```

Figure 42: Reporter's Viewpoint after Percolation - Model 2

viewpoint. The source list now shows the information about the first exposure to the belief, which has already been discussed, in addition to the new information about this second exposure to the belief. The names of the source and the reporter for this exposure as well as the ratings for the reporter's confidence in the source and the reporter's confidence in the reported information have been added to the source list. The reader's confidence in this new source and the reader's confidence in this new reporter have been evaluated. The ratings for both are NEUTRAL since the reader has no beliefs about the source or the reporter. The changes to the reader's viewpoint are shown in Figure 43.

As in Case 1, the next step would be to decompose the nesting of viewpoints since it is no longer needed. The only remaining viewpoints would be those of the system and the reader. After only two exposures to the belief ties(anti-abortion_extremists, radical_right_wing), the reader does not yet have enough confidence in the reliability of the belief to promote it to a held belief. The reliability of the belief will be re-evaluated each time the reader is exposed to it until the reliability of the belief reaches the reliability threshold and is promoted to a held belief.

```
System believes
  Reader believes the following:

          POTENTIAL Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
              Source 1: James Forester
              Reporter: Maggie Smith
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: HIGH
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL

              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL

            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                          far_right)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL

            play_footsie(anti-abortion_extremists,far_right)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL

            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL

            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
              Source 1: Warren Kinsella
              Reporter: Steve Mertl
                Reporter's confidence in the Source(s):      HIGH
                Reporter's confidence in the Information: NEUTRAL
                Reader's confidence in the Source(s):        NEUTRAL
                Reader's confidence in the Reporter:         NEUTRAL
```

Figure 43: Reader's Viewpoint after Percolation - Model 2

8.5.3 Case 3: Opposing Beliefs

The reader in this case is of the third model. The reader has previous opposing beliefs to the belief being percolated. As previously discussed, opposing held beliefs have an impact on the promotion mechanism (by increasing the reliability threshold) but they do not have an impact on the percolation mechanism. Potential opposing beliefs have no impact on the percolation mechanism either. A topic environment can contain two contradictory potential beliefs. However, it can not contain two contradictory held beliefs.

The initial state of the system for this case is as shown in Figure 37. The source, Warren Kinsella, has five beliefs in his anti-abortion topic environment. The reporter's viewpoint does not contain an anti-abortion topic environment since he has no previous beliefs about the topic. The reader's viewpoint does contain an anti-abortion topic environment since the reader does have previous beliefs about the topic. In fact, in this example, the reader's anti-abortion topic environment contains two opposing beliefs. One is the opposing held belief ~ties(anti-abortion_extremists,radical_right_wing). The other is the opposing potential belief ~fertile_recruiting_sites(anti-abortion_groups,hate_groups). As with all potential beliefs, this opposing potential belief has a source list that contains the information about the previous exposure to the belief. Here, the source list indicates the names of the reporter and the source involved in the previous exposure. It indicates that the reporter had NEUTRAL confidence in the source and LOW confidence in the reported information. The ratings for the reader's confidence in the source and for the reader's confidence in the reporter are both NEUTRAL since the reader has no previous beliefs about the source or the reporter.

```
System believes
  Reader believes
    Steve Mertl believes
      Warren Kinsella believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
            play_footsie(anti-abortion_extremists,far_right)
            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
```

Figure 44: Source's Viewpoint after Percolation - Model 3

Figures 44, 45, and 46 show the state of the viewpoints of the source, the reporter, and the reader, respectively, after percolation of the source's beliefs has occurred. Once again, the source's viewpoint has not changed. The reporter's viewpoint has changed as in Case 1 and Case 2. It now contains an anti-abortion topic environment into which the beliefs in the source's anti-abortion topic environment have been percolated. In addition, the source list for each belief has been started. At this point all that is known is that the source is Warren Kinsella.

```
System believes
  Reader believes
    Steve Mertl believes the following:

          Beliefs about anti-abortion
            ties(anti-abortion_extremists,radical_right_wing)
              Source 1: Warren Kinsella

            run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                           far_right)
              Source 1: Warren Kinsella

            play_footsie(anti-abortion_extremists,far_right)
              Source 1: Warren Kinsella

            use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
              Source 1: Warren Kinsella

            fertile_recruiting_sites(anti-abortion_groups,hate_groups)
              Source 1: Warren Kinsella
```

Figure 45: Reporter's Viewpoint after Percolation - Model 3

The reader's viewpoint also has changed. The beliefs in the reporter's anti-abortion topic environment have been percolated into the reader's topic environment. The three beliefs that have no opposing belief in the reader's topic environment were percolated as new potential beliefs. The belief ties(anti-abortion_extremists, radical_right_wing) which is opposed by the reader's held belief ~ties(anti-abortion_extremists, radical_right_wing) was also percolated as a new potential belief. Such a situation could be said to represent that the reader is beginning to have some doubts about the held belief. The reporter's belief fertile_recruiting_sites(anti-abortion_groups,hate_groups) opposed by the reader's potential belief ~fertile_recruiting_sites(anti-abortion_groups,hate_groups) is also percolated to the reader's anti-abortion topic environment as a new potential belief. That the reader has two opposing potential beliefs in the same topic environment does not cause a contradiction in *Percolator*. As previously mentioned, only held beliefs can participate in ascription activities. Each of the new potential beliefs has a source list containing the information relative to this exposure to the belief.

If the reader is exposed to the belief ties(anti-abortion_extremists, radical_right_wing) often enough, the reliability of the belief could reach the reliability threshold (which is higher when the reader has opposing beliefs). This would result in the promotion of the belief ties(anti-abortion_extremists, radical_right_wing) to the status of held belief. However, since a topic environment can not contain two contradictory held beliefs, the old belief ~ties(anti-abortion_extremists, radical_right_wing) would be removed from the topic environment and the belief maintenance mechanism (not implemented in *Percolator*) would be invoked to remove any beliefs that may have been dependent on the old belief.

In the case of the two opposing potential beliefs, if the reader is re-exposed to the beliefs, the reliability of one of the beliefs will reach the reliability threshold first and will be promoted to a held belief. The other belief will be retained as a potential belief. This will result in the situation in which a reader has a potential belief opposing a held belief. If the reader is exposed to the potential belief often enough, it will reach the reliability threshold and be promoted to a held belief. This will lead to another reversal of one of the reader's held beliefs.

To complete the modelling process, all Warren Kinsella's other beliefs as well as all the propositions made by other sources would need to be incorporated analogously. Once the propositions from the entire article have been incorporated into *Percolator*, the reader's viewpoint represents a model of the beliefs acquired by the reader from reading the news article. A web-search engine can then make decisions based on the reader's model about whether or not the article is appropriate for the reader.

A web-search engine can have many users. The reader modelling application will need to support models of many users. Applications can create their own default models for different types of readers, as was done in the example. Or, applications can maintain separate models for each user of the application. The example in this chapter has shown that *Percolator* can easily be used in a multi-model environment. Changing reader models is an extremely simple task.

```
System believes
   Reader believes the following:

         HELD Beliefs about anti-abortion
           ~ties(anti-abortion_extremists,radical_right_wing)

         POTENTIAL Beliefs about anti-abortion
           ~fertile_recruiting_sites(anti-abortion_groups,hate_groups)
             Source 1: President Heritage Front
             Reporter: Maggie Smith
               Reporter's confidence in the Source(s):   NEUTRAL
               Reporter's confidence in the Information: LOW
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL

           ties(anti-abortion_extremists,radical_right_wing)
             Source 1: Warren Kinsella
             Reporter: Steve Mertl
               Reporter's confidence in the Source(s):   HIGH
               Reporter's confidence in the Information: NEUTRAL
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL

           run_in_opposite_direction_from(a_lot_of_anti-abortion_extremists,
                                          far_right)
             Source 1: Warren Kinsella
             Reporter: Steve Mertl
               Reporter's confidence in the Source(s):   HIGH
               Reporter's confidence in the Information: NEUTRAL
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL

           play_footsie(anti-abortion_extremists,far_right)
             Source 1: Warren Kinsella
             Reporter: Steve Mertl
               Reporter's confidence in the Source(s):   HIGH
               Reporter's confidence in the Information: NEUTRAL
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL

           use(hate_groups,anti-abortion_stuff,boost_membership_attract_people)
             Source 1: Warren Kinsella
             Reporter: Steve Mertl
               Reporter's confidence in the Source(s):   HIGH
               Reporter's confidence in the Information: NEUTRAL
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL

           fertile_recruiting_sites(anti-abortion_groups,hate_groups)
             Source 1: Warren Kinsella
             Reporter: Steve Mertl
               Reporter's confidence in the Source(s):   HIGH
               Reporter's confidence in the Information: NEUTRAL
               Reader's confidence in the Source(s):     NEUTRAL
               Reader's confidence in the Reporter:      NEUTRAL
```

Figure 46: Reader's Viewpoint after Percolation - Model 3

8.6 Summary

In summary, this thesis outlines several concepts that can be combined to create a system that builds user models by simulating how a reader acquires beliefs from news articles.

It was shown how a news article can be represented as a ***complex profile structure*** separating the parts of the ***reported speech*** – the proposition, the reporting verb, and the lexicalisation of the source. By performing ***evidential analysis*** on the reporting verb and on the lexicalisation of the source, the reporter's confidence in the reported information and the reporter's confidence in the source can be evaluated.

Viewpoints can then be generated for each agent involved with the news article – the system, the reader, the reporter, and all of the sources mentioned in the article. ***Belief ascription*** can be used to populate the viewpoints of the different agents (although this was not done in the implementation of *Percolator*). The propositions extracted from the profiling of the article can then be input into the source's viewpoint as ***held beliefs*** which are then ***percolated*** through the viewpoint nesting to the reader's viewpoint as ***potential beliefs*** in the appropriate ***topic environments***.

A ***source list*** is created for each belief providing some ***justification*** for the belief. The reader's confidence in the source and the reader's confidence in the reporter are evaluated based on the reader's current set of beliefs. This information along with the names of the reporter and source(s) as well as the evaluation of the reporter's confidence in the source and the reporter's confidence in the reported information are stored in the source list for each belief. The source list is augmented with each exposure to the belief.

A ***belief heuristic*** uses the four confidence criteria to determine whether or not the reader has sufficient confidence in the reliability of the belief to accept it as true. When the reliability is sufficient, ***belief promotion*** is performed and the potential belief becomes a held belief. When a contradiction in the held beliefs of the reader is detected, the old belief is retracted in favour of the new belief. ***Belief revision*** is then performed on the nested viewpoints to retract or revise any beliefs that were dependent on the old belief.

The solution presented in this thesis can be used to create an agent that would search the Internet on behalf of the reader filtering out documents that are of little interest to the reader based on the model of the reader's beliefs.

Bibliography

[Alchourrón *et al.*, 1985] C. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction functions and their associated revision functions. *Journal of Symbolic Logic*, 50:510–530, 1985.

[Bacchus *et al.*, 1994] F. Bacchus, A. J. Grove, J. Y. Halpern, and D. Koller. Generating new beliefs from old. In *Uncertainty in Artificial Intelligence, Proceedings of the Tenth Conference*, 1994.

[Ballim and Wilks, 1991] A. Ballim and Y. Wilks. Beliefs, sterotypes and dynamic agent modelling. *User Modelling and User-Adapted Interaction*, 1:33–65, 1991.

[Ballim and Wilks, 1992] A. Ballim and Y. Wilks. *Artificial Believers*. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1992.

[Ballim *et al.*, 1991] A. Ballim, Y. Wilks, and J.A. Barnden. Belief ascription, metaphor, and intensional identification. *Cognitive Science*, 15:133–171, 1991.

[Ballim, 1987] A. Ballim. The subjective ascription of belief to agents. In J. Hallam and C. Mellish, editors, *Advances in Artificial Intelligence*, pages 267–278. Wiley and Sons, 1987.

[Ballim, 1992] A. Ballim. *ViewFinder: A Framework for Representing, Ascribing and Maintaining Nested Beliefs of Interacting Agents*. PhD thesis, Université de Genève, 1992.

[Bergler, 1992] S. Bergler. *Evidential Analysis of Reported Speech*. PhD thesis, Brandeis University, 1992.

[Bergler, 1993] S. Bergler. Semantic dimensions in the field of reporting verbs. In *Making Sense of Words, Proceedings of the Ninth Annual Conference of the UW Centre for the New OED and Text Research*, pages 44–56, Oxford, U.K., September 1993.

[Bergler, 1995a] S. Bergler. From lexical semantics to text analysis. In P. Saint-Dizier and E. Viegas, editors, *Computational Lexical Semantics*, pages 98–124. Cambridge University Press, 1995.

[Bergler, 1995b] S. Bergler. Generative lexicon principles for machine translation: A case for metalexical structure. *Journal of Machine Translation*, 9(3), 1995.

[Boutilier and Becher, 1995] C. Boutilier and V. Becher. Abduction as belief revision. *Artificial Intelligence*, 77:43–94, 1995.

[Bradshaw, 1997] J. Bradshaw. Introduction. In J. Bradshaw, editor, *Software Agents.* AAAI Press/The MIT Press, 1997.

[Chin, 1991] D. N. Chin. Intelligent interfaces as agents. In J. Sullivan and S. Tyler, editors, *Intelligent User Interfaces*, chapter 9, pages 177–206. ACM Press, New York, N.Y., 1991.

[de Kleer, 1986] J. de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28(2):127–162, 1986.

[Doyle, 1979] J. Doyle. A truth maintenance system. *Artificial Intelligence*, 12(2):231–272, 1979.

[Doyle, 1983] J. Doyle. The ins and outs of reason maintenance. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 349–351, 1983.

[Doyle, 1991] J. Doyle. Rational belief revision (preliminary report). In *Proceedings of the Second Conference on Principles of Knowledge Representation and Reasoning*, San Mateo, California, 1991.

[Doyle, 1992] J. Doyle. Reason maintenance and belief revision: Foundation vs. coherence theories. In P. Gärdenfors, editor, *Belief Revision*, pages 29–51. Cambridge University Press, Cambridge, MA, 1992.

[Everett and Forbus, 1996] J. O. Everett and K. D. Forbus. Scaling up logic-based truth maintenance systems via fact garbage collection. In *Proceedings of the 13th National Conference on Artificial Intelligence*, 1996.

[Fagin and Halpern, 1985] R. Fagin and J. Y. Halpern. Belief, awareness, and limited reasoning. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pages 491–501, 1985.

[Falkenhainer, 1986] B. Falkenhainer. Towards a general purpose belief maintenance system. In *Proceedings of Second Workshop on Uncertainty in AI*, pages 71–76, 1986.

[Florea, 1997] A. M. Florea. Assumption-based reasoning of intelligent agents. In *Proceedings of the Eleventh International Conference on Control Systems and Computer Science*, pages 126–133, Bucharest, May 28-29 1997.

[Friedman and Halpern, 1994] N. Friedman and J. Y. Halpern. Conditional logics of belief change. In *Proccedings of the National Conference on Artificial Intelligence*, pages 915–921, 1994.

[Gärdenfors and Rott, 1995] P. Gärdenfors and H. Rott. Belief revision. In D. M. Gabbay, C. J. Hogger, and J. A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 4, pages 35–132. Clarendon Press, Oxford, 1995.

[Gärdenfors, 1988] P. Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States.* MIT Press, Cambridge, MA, 1988.

[Georgeff *et al.*, 1999] M. Georgeff, B. Pell, M. Pollack, M. Tambe, and M. Wooldridge. The belief-desire-intention model of agency. In *Proceedings of Agents, Theories, Architectures and Languages*, 1999.

[Hintikka, 1962] J. Hintikka. *Knowledge and Belief.* Cornell University Press, 1962.

[Hintikka, 1975] J. Hintikka. Impossible possible worlds vindicated. *Journal of Philosophical Logic*, 4, 1975.

[Isozaki and Katsuno, 1996] H. Isozaki and H. Katsuno. A semantic characterization of an algorithm for estimating others' beliefs from observation. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Conference on Innovative Applications of Artificial Intelligence*, volume 1, 1996.

[Johnson and Shapiro, 2000] F. L Johnson and S. C. Shapiro. Implementing integrity constraints in an existing belief revision model: CSE Technical Report 2000-03. In *Proceedings of the International Workshop on Belief Change and the Journal of Applied Non-Classical Logics*, 2000.

[Kaminka and Tambe, 1998] G. A. Kaminka and M. Tambe. What is wrong with us? Improving robustness through social diagnosis. In *Proceedings of the 15th Conference on Artificial Intelligence*, Madison, Wisconsin, July 26-30 1998.

[Kaminka *et al.*, 1998] G. A. Kaminka, M. Tambe, and C. M. Hopper. The role of agent-modeling in agent robustness. In *AI Meets the Real-World: Lessons Learned*, Stanford, CT, September 1998.

[Konolige, 1986] K. Konolige. What awareness isn't: A sentential view of implicit and explicit belief. In Joseph Y. Halpern, editor, *Theoretical Aspects of Reasoning About Knowledge, Proceedings of the 1986 Conference*, pages 241–250, 1986.

[Lee and Wilks, 1996] M. Lee and Y. Wilks. An ascription-based approach to speech acts. In *Proceedings of the 16th Conference on Computational Linguistics*, Copenhagen, 1996.

[Levesque, 1984] H.J. Levesque. A logic of implicit and explicit beliefs. In *Proceedings of the National Conference on Artificial Intelligence*, pages 198–202, 1984.

[Li and Pereira, 1996] R. Li and L. M. Pereira. What is believed is what is explained (sometimes). In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Conference on Innovative Applications of Artificial Intelligence*, volume 1, pages 550–555, Portland, Oregon, 1996.

[Liberatore and Schaerf, 1996] P. Liberatore and M. Schaerf. The complexity of model checking for belief revision and update. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Conference on Innovative Applications of Artificial Intelligence*, volume 1, 1996.

[Maes, 1995] P. Maes. Modeling adaptive autonomous agents. In C. G. Langton, editor, *Artificial Life, An Overview*. MIT Press, 1995.

[Martins and Shapiro, 1986] J. P. Martins and S. C. Shapiro. Theoretical foundations for belief revision. In *Theoretical Aspects of Reasoning about Knowledge, Proceedings of the 1986 Conference*, 1986.

[McAllester, 1978] D. McAllester. A three valued truth maintenance system. Master's thesis, Massachusetts Institute of Technology, 1978.

[McAllester, 1990] D. McAllester. Truth maintenance. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 1109–1115, Boston, MA, 1990.

[Minsky, 1975] M. Minsky. A framework for representing knowledge. In P. Winston, editor, *Psychology of Computer Vision*. MIT Press, Cambridge, MA, 1975.

[Moukas, 1997] A. Moukas. *Amalthea*: Information discovery and filtering using a multiagent evolving ecosystem. *Applied Artificial Intelligence: An International Journal*, 11(5):437–457, 1997.

[Nebel, 1989] B. Nebel. A knowlege level analysis of belief revision. In R. J. Brachman, H. J. Levesque, and R. Reiter, editors, *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*, pages 301–311, San Mateo, CA, May 1989. Morgan Kaufmann.

[Pollack, 1987] M. E. Pollack. A model of plan inference that distinguishes between the beliefs of actors and observers. In M. Georgeff and A. Lansky, editors, *The 1986 Workshop on Reasoning about Actions and Plans*, pages 207–214, Los Altos, CA, 1987. Morgan Kaufmann.

[Pustejovsky, 1991] J. Pustejovsky. The Generative Lexicon. *Computational Linguistics*, 17(4), 1991.

[Ramoni and Riva, 1993] M. Ramoni and A. Riva. Belief maintenance with probabilistic logic. In *Proceedings of the AAAI Fall Symposium on Automated Deduction in Non Standard Logics*, Raleigh, N.C., 1993.

[Ramoni and Riva, 1994] M. Ramoni and A. Riva. Belief maintenance in Bayesian networks. In *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence*, pages 498–505, Seattle, Washington, July 29-31 1994.

[Rich, 1989] E. Rich. Stereotypes and user modeling. *User Models in Dialog Systems*, 1989.

[Schank and Abelson, 1977] R. Schank and R. Abelson. *Scripts, Plans, Goals, and Understanding*. Lawrence Erlbaum Associates Inc., Hillsdale, New Jersey, 1977.

[Sidner and Israel, 1981] C. L. Sidner and D. J. Israel. Recognizing intended meaning and speakers' plans. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, volume 1, 1981.

[Snow, 1994] P. Snow. The emergence of ordered belief from initial ignorance. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, volume 1, 1994.

[Taylor and Whitehill, 1981] G.B. Taylor and S.B. Whitehill. A belief representation for understanding deception. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, Vancouver, August 24–28 1981.

[Vardi, 1986] M. Y. Vardi. On epistemic logic and logical omnicience. In J. Y. Halpern, editor, *Theoretical Aspects of Reasoning about Knowledge, Proceedings of the 1986 Conference*, pages 293–305, 1986.

[Weydert, 1994] E. Weydert. General belief measures. In *Uncertainty in Artificial Intelligence, Proceedings of the Tenth Conference*, 1994.

[Wiebe, 1990] J. Wiebe. Identifying subjective characters in narrative. In *Proceedings of the 13th International Conference on Computational Linguistics*, Helsinki, 1990.

[Wilks and Ballim, 1987] Y. Wilks and A. Ballim. Multiple agents and the heuristic ascription of belief. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pages 118–124, 1987.

[Wilks and Bien, 1983] Y. Wilks and J. Bien. Beliefs, points of view, and multiple environments. *Cognitive Science*, 7(1):95–119, 1983.

[Wilks, 1986] Y. Wilks. Relevance and beliefs. In T. Myers, K. Brown, and B. McGonigle, editors, *Reasoning and Discourse Processes*, pages 265–289. Academic Press, New York, 1986.

[Witte, 1994] R. Witte. A profiling system for newspaper articles. Technical report, Concordia University Computer Science, 1994.

[Witteveen and Brewka, 1993] C. Witteveen and G. Brewka. Skeptical reason maintenance and belief revision. *Artificial Intelligence*, 61:1–36, 1993.