Preprint of an article accepted for publication in
International Journal on Artificial Intelligence Tools

http://www.worldscientific.com/worldscinet/ijait

International Journal on Artificial Intelligence Tools


# A Theoretical Framework for Trust-based News Recommender Systems and its Implementation using Defeasible Argumentation

Cristian E. Briguez
*Departamento de Ciencias e Ingeniería de la Computación,*
*Universidad Nacional del Sur, Av. Alem 1253, (8000) Bahía Blanca, Argentina*
*ceb@cs.uns.edu.ar*

Marcela Capobianco
*Departamento de Ciencias e Ingeniería de la Computación,*
*Universidad Nacional del Sur, Av. Alem 1253, (8000) Bahía Blanca, Argentina*
*mc@cs.uns.edu.ar*

Ana G. Maguitman
*Departamento de Ciencias e Ingeniería de la Computación,*
*Universidad Nacional del Sur, Av. Alem 1253, (8000) Bahía Blanca, Argentina*
*agm@cs.uns.edu.ar*

Although the importance of trust in recommender systems is widely recognized, the actual mechanisms of trust propagation and trust preservation are poorly understood. This is partly due to the fact that trust is a complex notion, which is typically context dependent, subjective, dynamic and not always transitive or symmetrical. This paper presents a theoretical analysis of the notion of trust in news recommendation and discusses the advantages of modeling this notion using Defeasible Logic Programming, a general-purpose defeasible argumentation formalism based on logic programming. In the proposed framework, users can express explicit trust statements on news reports, news sources and other users. Trust is then modeled and propagated using a dialectical process supported by a Defeasible Logic Programming interpreter. A set of basic postulates for trust and their representation by means of defeasible rules is presented. The suitability of the approach is investigated with a set of illustrative examples and then analyzed from a formal perspective. The obtained results indicate that the proposed framework provides a solid foundation for building trust-based news recommendation services.

*Keywords*: Argumentation, News Recommender Systems, Trust Management.

## 1. Introduction

We frequently seek suggestions from people we trust for deciding the best place to acquire some service or the best source to obtain information about a certain topic. Suggestions of people we trust may also help decide who else to trust. The Web offers new opportunities to create recommendation services based on trust. In particular, news management systems on the Web can take advantage of the large community of readers to rank news, determine the reputation of an information source or propagate trust among users. This can help decide which news are more interesting or trustworthy to a certain user, providing more personalized services.

The dynamics of news credibility has mainly been studied through quantitative approaches (e.g. [28]). However, a purely quantitative perspective to news credibility has several limitations. In particular, quantitative approaches make it hard to provide readers with a justification of why certain news should be trusted, or might be unable to deal with subtle notions such as distrust or trust repair.

A more appealing approach would consist in combining quantitative and qualitative criteria to filter and rank news. In this sense, quantitative methods can help determine if the news topic is relevant to the user's interest, while qualitative criteria could help decide whether the news comes from a reliable source. For example, a qualitative approach is more natural to deal with some properties of trust such as being context dependent, subjective, asymmetrical, dynamic and not always transitive.

This paper extends previous work on recommendation technologies presented in [16] by introducing new postulates for modeling trust propagation and by performing a theoretical investigation of trust-based recommendation systems for web news. The theoretical investigation is performed by identifying a set of desirable properties of trust and by providing a formal proof of two important properties of the model, namely the fact that the premises that establish a viewer's credibility status regarding a news report cannot be revoked (*inclusion*), and the fact that the system cannot conclude that an entity is simultaneously trusted and distrusted by the same viewer (*consistency*).

A key ingredient of the proposed model is the propagation of trust based on inference mechanisms. In particular, a defeasible logic programming interpreter is used to manage interpersonal trust and distrust. Trust is modeled by means of twelve basic postulates, which can be easily extended or relaxed. The postulates are translated into defeasible rules, which allow to infer tentative conclusions regarding the credibility of a news report by a reader. In addition to the theoretical analysis, we demonstrate the use of the tool with a set of illustrative examples, outline the current system implementation and discuss future work.

The main contribution of this work is a deep study of the notion of trust, performed by a careful theoretical analysis of its properties. The article puts special emphasis in discussing the suitability of Defeasible Logic Programming as the underlying framework for a computational model of trust propagation. Besides introducing a number of theoretical results, we discuss several examples and present the outcomes of a simulation which help provide additional evidence supporting the appropriateness of the proposed framework for modeling the notion of trust.

## 2. Background

In this section we first discuss previous work on news recommendation systems based on trust propagation and next we present the argumentative formalism of Defeasible

Logic Programming that will be used as the underlying logic for propagating trust statements.

### 2.1. *Web news and trust*

Web news services are especially attractive because, differently from printed newspapers, online news can be delivered in a rapid and personalized manner. Multi-source news providers on the Web, such as Google News [1], Yahoo! News [3], and MSNBC [4], integrate news from several sources and deliver them ranked based on factors such as news popularity, sources, freshness, reader's location and reader's interests.

Developing an algorithm for web news selection and ranking is very difficult because it requires combining many, sometimes conflicting, aspects. As a consequence, the problem of ranking and recommending web news has attracted much research attention in recent years. There are several reasons why measures of page authority such as PageRank [10] cannot be directly applied at the moment of ranking web news. Differently from what happens with web pages, the Internet newspapers rarely use linking. Moreover, breaking news usually have priority over previous news because viewers prefer to see information about news events as soon as they take place. Nevertheless, fresh news usually have very few incoming links, which precludes the application of link analysis algorithms to favor fresh news over stale ones. A ranking model that gives high priority to fresh news, however, will have some deficiencies. Reports on fresh news tend to be incomplete and many stories presented as breaking news are revised when additional information becomes available.

Another approach to rank news could be based on news popularity, estimated by monitoring the number of viewers accessing a report or by a system of voting on favorite stories. However, news popularity may not reflect the real value a news has for individual viewers.

Usually a viewer has to decide whether a news report is worth reading and whether the facts described in the report are credible. The information provided in news reports may not always be fully verifiable and therefore another important factor that can help select news is *trust* or *credibility.* The level of trust a viewer has on a piece of news is not necessarily associated with measures of news authority or popularity, and it may even be negatively correlated with news freshness. A news recommendation service that uses a trust-management system can support the viewer in making the decision by selecting reports from trusted sources or based on other trustworthy viewers' opinions.

Trust is a fundamental concept in human behavior, which for many years has enabled collaboration. Therefore, trust is an important aspect in the implementation of recommendation systems in general, and web news recommenders in particular. Typically, the notion of trust is defined in terms of two components: trusting intentions and trusting beliefs. For example, a user can trust the intentions of a vendor or the intentions of a service or information provider. On the other hand, a user can trust the beliefs of other users.

Research in trust modeling has focused mainly on computing the amount of trust an agent will have on another agent if there is not an explicit trust relation between them (e.g., [7,20,23]). In most of these approaches, trust values are represented within the range [-1, 1]. On the other hand, formal characterizations of the notion of trust have been investigated using logics and semantics of trust (e.g., [14,22,27]), while argumentation has been used to reason about trust in [35,31]. A trust model that applies temporal defeasible logic is presented in [25]. The core of this approach is to efficiently compute a reputation model based on agents's direct experience and reports provided by others.

Trust models have been applied in a number of areas, such as E-commerce [29], Social Networks [36] and P2P systems [17]. Models of trust have also been applied to the area of web news recommendation. For example, in [28] a method is proposed to rate the credibility of news documents using algorithms that compare the content of different news sources. PolyNews [30] is a news service framework that tries to mitigate the effect of media bias by the creation of multiple classified viewpoints. NewsTrust [2] is a service created to evaluate news where users can rank news reports, news writers and news sources. A multi-layer recommendation system for publications based on trust is proposed in [21]. Another system, PersoNews [6], utilizes a Naïve Bayes classifier to provide a personalized RSS reader. Differently from our own proposal, these approaches model the notions of news credibility and media bias through quantitative mechanisms. An argument-based approach for generating recommendations by agents is proposed in [9]. Although this work relates to our proposal in using defeasible rules to generate recommendations, it does not address the problem of news recommendation and focuses on ambient intelligence environments, without attempting to model the notion of trust.

### 2.2. *Defeasible logic programming*

Defeasible logic programming (DeLP) [19] is a general-purpose defeasible argumentation formalism based on logic programming, intended to model inconsistent and potentially contradictory knowledge. This formalism provides a knowledge representation language which gives the possibility of representing tentative information in a declarative manner, and a reasoning mechanism that considers every way a conclusion could be supported and decides which one has the best support.

An interesting feature of DeLP is that conclusions obtained trough this formalism can be easily explained by the argumentative reasoning process. Also, DeLP deals with fallacious reasoning, solving situations that are problematic for many argumentation formalisms, as shown in Garcia and Simari's work [19]. Also, DeLP has the advantage of being fully implemented and of providing a powerful argument visualization module very useful for generating examples and test beds.

In DeLP, a defeasible logic program has the form $\mathcal{P} = (\Pi, \Delta)$, where $\Pi$ and $\Delta$ stand for *strict* and *defeasible* knowledge, respectively. The set $\Pi$ involves *strict rules* of the form $P \leftarrow Q_1, \ldots, Q_k$ and *facts* (strict rules with empty body), and it is assumed to be *non-contradictory* (i.e., no complementary literals $P$ and $\sim P$ can be inferred, where $\sim P$ denotes the contrary of $P$). The set $\Delta$ involves *defeasible rules* of the form $P \prec Q_1, \ldots, Q_k$, which stand for $Q_1, \ldots, Q_k$ provide a tentative reason to believe $P$. Rules in DeLP are defined in terms of *literals*. A literal is an atom $A$ or the strict negation ($\sim A$) of an atom. Default negation (denoted not $A$) is also allowed in the body of defeasible rules (see [19] for details).

Deriving literals in DeLP results in the construction of *arguments*.

**Definition 2.1.** *[Defeasible Derivation]* Let $\mathcal{P} = (\Pi, \Delta)$ be a DeLP program and let $Q$ be a ground literal. A finite sequence of ground literals,

$$s = Q_1, Q_2, \ldots, Q_{n-1}, Q$$

is said to be a *defeasible derivation* for $Q$ from $\mathcal{P}$ (abbreviated $\mathcal{P} \mathrel{|\!\!\sim} Q$) if for every $Q_i$, $1 \leq i \leq n$,

(1) the literal $Q_i$ can be derived from $\Pi$, or
(2) there exists a defeasible rule $r \in \Delta$ and a ground instance $t$ of $r$, $t = Q_i \prec L_1, \ldots, L_m$, where $L_1, \ldots, L_m$ are ground literals previously occurring in the sequence $s$.

**Definition 2.2.** *[Argument – Sub-argument]* Given a DeLP program $\mathcal{P}$, an *argument* $\mathcal{A}$ for a ground literal $Q$, also denoted $\langle \mathcal{A}, Q \rangle$, is a subset of ground instances of the defeasible rules in $\mathcal{P}$ such that:

(1) there exists a defeasible derivation for $Q$ from $\Pi \cup \mathcal{A}$,
(2) $\Pi \cup \mathcal{A}$ is non-contradictory,
(3) There is no $\mathcal{A}' \subset \mathcal{A}$ such that $\Pi \cup \mathcal{A}' \mathrel{|\!\!\sim} Q$.

Given two arguments $\langle \mathcal{A}_1, Q_1 \rangle$ and $\langle \mathcal{A}_2, Q_2 \rangle$, we will say that $\langle \mathcal{A}_1, Q_1 \rangle$ is a *sub-argument* of $\langle \mathcal{A}_2, Q_2 \rangle$ iff $\mathcal{A}_1 \subseteq \mathcal{A}_2$.

Note that arguments are obtained by a mechanism similar to the usual query-driven SLD derivation from logic programming, performed by backward chaining on *both* strict and defeasible rules; in this context a negated literal $\sim P$ is treated just as a new predicate name *no* $P$. In DeLP, arguments provide tentative support for claims (literals). Clearly, as a program $\mathcal{P}$ represents incomplete and tentative information, an argument $\langle \mathcal{A}, Q \rangle$ may be *attacked* by other arguments also derivable from $\mathcal{P}$. An argument $\langle \mathcal{B}, R \rangle$ is a *counter-argument* for $\langle \mathcal{A}, Q \rangle$ whenever a subargument $\langle \mathcal{A}', Q' \rangle$ (with $\mathcal{A}' \subseteq \mathcal{A}$) in $\langle \mathcal{A}, Q \rangle$ can be identified, such that $\langle \mathcal{B}, R \rangle$ and $\langle \mathcal{A}', Q' \rangle$ cannot be simultaneously accepted since their joint acceptance would allow contradictory conclusions to be inferred from $\Pi \cup \mathcal{A}' \cup \mathcal{B}$. If the attacking argument $\langle \mathcal{B}, R \rangle$ is preferred over $\langle \mathcal{A}', Q' \rangle$, then $\langle \mathcal{B}, R \rangle$ is called a *defeater* for $\langle \mathcal{A}, Q \rangle$. The preference criterion commonly used is *specificity* [19], preferring those arguments which are more direct or more informed, although other criteria could be adopted. Defeaters are also classified into proper defeaters and blocking defeaters. An argument

$\langle \mathcal{A}, Q \rangle$ is a proper defeater for an argument $\langle \mathcal{B}, R \rangle$ according to a given preference criterion "$\preceq$" if $\langle \mathcal{A}, Q \rangle$ is preferred to $\langle \mathcal{B}, R \rangle$ according to "$\preceq$". On the contrary, $\langle \mathcal{A}, Q \rangle$ is a blocking defeater of $\langle \mathcal{B}, R \rangle$ if $\langle \mathcal{A}, Q \rangle$ is unrelated to $\langle \mathcal{B}, R \rangle$ according to "$\preceq$".

In DeLP the search for defeaters for a given argument $\langle \mathcal{A}, Q \rangle$ prompts for a recursive process, resulting in the generation of a *dialectical tree*: the root node of this tree is the original argument at issue, and every children node in the tree is a defeater for its parent.

**Definition 2.3.** *[Dialectical Tree]* The *dialectical tree* for an argument $\langle \mathcal{A}, Q \rangle$, denoted $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$, is recursively defined as follows:

(1) A single node labeled with an argument $\langle \mathcal{A}, Q \rangle$ with no defeaters (proper or blocking) is by itself the dialectical tree for $\langle \mathcal{A}, Q \rangle$.
(2) Let $\langle \mathcal{A}_1, Q_1 \rangle, \langle \mathcal{A}_2, Q_2 \rangle, \ldots, \langle \mathcal{A}_n, Q_n \rangle$ be all the defeaters (proper or blocking) for $\langle \mathcal{A}, Q \rangle$. The dialectical tree for $\langle \mathcal{A}, Q \rangle$, $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$, is obtained by labeling the root node with $\langle \mathcal{A}, Q \rangle$, and making this node the parent of the root nodes for the dialectical trees of $\langle \mathcal{A}_1, Q_1 \rangle, \langle \mathcal{A}_2, Q_2 \rangle, \ldots, \langle \mathcal{A}_n, Q_n \rangle$

The marking of the dialectical tree is performed as in an AND-OR trees: leaves are always marked as undefeated nodes (as they have no defeaters); inner nodes can be be marked either as undefeated (if and only if every of its children nodes is marked as defeated) or as defeated (whenever at least one of its children has been marked as undefeated). The original argument $\langle \mathcal{A}, Q \rangle$ (the root of tree) is deemed as ultimately acceptable or *warranted* whenever it turns out to be marked as undefeated after applying the above process.

**Definition 2.4.** *[Marking of the Dialectical Tree]* Let $\langle \mathcal{A}_1, Q_1 \rangle$ be an argument and $\mathcal{T}_{\langle \mathcal{A}_1, Q_1 \rangle}$ its dialectical tree, then:

(1) All the leaves in $\mathcal{T}_{\langle \mathcal{A}_1, Q_1 \rangle}$ are marked as a `U-node`.
(2) Let $\langle \mathcal{A}_2, Q_2 \rangle$ be an inner node of $\mathcal{T}_{\langle \mathcal{A}_1, Q_1 \rangle}$. Then $\langle \mathcal{A}_2, Q_2 \rangle$ is marked as a `U-node` iff every child of $\langle \mathcal{A}_2, Q_2 \rangle$ is marked as a `D-node`. The node $\langle \mathcal{A}_2, Q_2 \rangle$ is marked as a `D-node` if and only if it has at least a child marked as a `U-node`.

Dialectical analysis may in some situations give rise to fallacious argumentation. In DeLP dialectical trees are ensured to be free of fallacies by applying additional constraints when building *argumentation lines* (the different possible paths in a dialectical tree). The notions that follow have been developed to address these issues.

**Definition 2.5.** *[Argumentation line]*
Let $\mathcal{P} = (\Pi, \Delta)$ be a DeLP program and let $\langle \mathcal{A}, Q \rangle$ be an argument wrt $\mathcal{P}$. An *argumentation line* starting from $\langle \mathcal{A}, Q \rangle$, denoted $\lambda^{\langle \mathcal{A}, Q \rangle}$ (or simply $\lambda$), is a possibly infinite sequence of arguments

$$\lambda^{\langle \mathcal{A}, Q \rangle} = [\langle \mathcal{A}_0, Q_0 \rangle, \langle \mathcal{A}_1, Q_1 \rangle, \ldots, \langle \mathcal{A}_n, Q_n \rangle, \ldots]$$

satisfying the following conditions:

(1) If $\langle \mathcal{A}, Q\rangle$ has no defeaters, then $\lambda^{\langle \mathcal{A},Q\rangle} = [\langle \mathcal{A}, Q\rangle]$.
(2) If $\langle \mathcal{A}, Q\rangle$ has a defeater $\langle \mathcal{B}, P\rangle$ in $\mathcal{P}$, then $\lambda^{\langle \mathcal{A},Q\rangle} = \langle \mathcal{A}, Q\rangle \circ \lambda^{\langle \mathcal{B},R\rangle}$.

where the '$\circ$' operator stands for adding $\langle \mathcal{A}, Q\rangle$ as the first element of $\lambda^{\langle \mathcal{B},P\rangle}$.

In each argumentation line $\lambda^{\langle \mathcal{A},Q\rangle} = [\langle \mathcal{A}_0, Q_0\rangle, \langle \mathcal{A}_1, Q_1\rangle, \ldots, \langle \mathcal{A}_n, Q_n\rangle, \ldots]$ the argument $\langle \mathcal{A}_0, Q_0\rangle$ is supporting the main query $Q_0$, and every argument $\langle \mathcal{A}_i, Q_i\rangle$ defeats its predecessor $\langle \mathcal{A}_{i-1}, Q_{i-1}\rangle$. Thus, for $k \geq 0$, $\langle \mathcal{A}_{2k}, Q_{2k}\rangle$ is a supporting argument for $Q_0$ and $\langle \mathcal{A}_{2k+1}, Q_{2k+1}\rangle$ is an interfering argument for $Q_0$. In other words, every argument in the line supports $Q_0$ or interferes with it. As a result, an argumentation line can be split in two disjoint sets: $\lambda_S$ of supporting arguments, and $\lambda_I$ of interfering arguments.

On the basis of the above notions, fallacies that could appear in argumentation lines in DeLP programs can be classified as follows:

(1) An argument $\mathcal{A}_1$ could be introduced in an argumentation line *both as an interfering and supporting argument*, producing a *contradictory* argumentation line *e.g.*, $\lambda_1 = [\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_1, \ldots]$.
(2) An argument $\mathcal{A}_1$ could be reintroduced as a supporting argument *for itself*. In that case a *circular* argumentation line would result, *e.g.*, $\lambda_2 = [\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_1, \ldots]$.

Argumentation lines as $\lambda_1$ and $\lambda_2$ should not be considered as acceptable, as they represent flawed reasoning processes. These fallacious situations can be generalized to cycles of any length: even cycles evidence contradictory argumentation, whereas odd cycles indicate circular argumentation. To solve these problems the following concepts are introduced:

**Definition 2.6.** Contradictory set of arguments
A set of arguments $S = \bigcup_{i=1}^{n}\{\langle \mathcal{A}_i, Q_i\rangle\}$ is *contradictory* with respect to a program $\mathcal{P} = (\Pi, \Delta)$ if and only if the set $\Pi \cup \bigcup_{i=1}^{n} \mathcal{A}_i$ allows the derivation of complementary literals.

**Definition 2.7.** Acceptable argumentation line
Let $\mathcal{P} = (\Pi, \Delta)$ be a program, and let

$$\lambda = [\langle \mathcal{A}_0, q_0\rangle, \langle \mathcal{A}_1, Q_1\rangle, \ldots, \langle \mathcal{A}_n, Q_n\rangle, \ldots]$$

be an argumentation line in $\mathcal{P}$, such that

$$\lambda' = [\langle \mathcal{A}_0, Q_0\rangle, \langle \mathcal{A}_1, Q_1\rangle, \ldots, \langle \mathcal{A}_k, Q_k\rangle]$$

is an initial segment of $\lambda$. The sequence $\lambda'$ is an *acceptable argumentation line* in $\mathcal{P}$ if and only if it is the longest initial segment in $\lambda$ satisfying the following conditions:

(1) The sets $\lambda'_S$ and $\lambda'_I$ are each non-contradictory sets of arguments with respect to $\mathcal{P}$.

(2) No argument $\langle \mathcal{A}_j, Q_j \rangle$ in $\lambda'$ is a sub-argument of an argument $\langle \mathcal{A}_i, Q_i \rangle$ of $\lambda'$, $i < j$.

(3) In $\lambda'$ there is no subsequence of arguments

$$[\langle \mathcal{A}_{i-1}, Q_{i-1} \rangle, \langle \mathcal{A}_i, Q_i \rangle, \langle \mathcal{A}_{i+1}, Q_{i+1} \rangle],$$

such that $\langle \mathcal{A}_i, Q_i \rangle$ is a blocking defeater for $\langle \mathcal{A}_{i-1}, Q_{i-1} \rangle$, and $\langle \mathcal{A}_{i+1}, Q_{i+1} \rangle$ is a blocking defeater for $\langle \mathcal{A}_i, Q_i \rangle$.

Let us analyze the rationale for the conditions in Definition 2.7. Condition 1 prohibits the use of contradictory information on either side (proponent or opponent). Condition 2 eliminates circular reasoning. Finally, condition 3 enforces the use of a stronger argument to defeat an argument which acts as a blocking defeater. The reason for this policy is justified by the following considerations. Suppose that argumentation lines containing two consecutive blocking defeaters were allowed, and consider the following scenario. An argument $\langle \mathcal{A}, L \rangle$ is blocked by $\langle \mathcal{B}, \sim L \rangle$ which in turn is blocked by $\langle \mathcal{C}, L \rangle$. If there are no more arguments to take into account, $\langle \mathcal{A}, L \rangle$ would be warranted. Nevertheless, the support for $L$ is no better than the support for $\sim L$.

An *acceptable dialectical tree* is defined in turn as a tree where every argumentation line is *acceptable* and the notion of warrant in DeLP is grounded on this concept. Given a query $Q$ and a DeLP program $\mathcal{P}$, we will say that $Q$ is *warranted* wrt $\mathcal{P}$ iff there exists an argument $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$ such that the root of its associated dialectical tree $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$ is marked as a $U$-node.

**Definition 2.8.** *[Warrant]* Let $\mathcal{A}$ be an argument for a literal $Q$, and let $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$ be its associated dialectical tree. Argument $\mathcal{A}$ is a *warrant* for $Q$ if and only if the root of $\mathcal{T}_{\langle \mathcal{A}, Q \rangle}$ is marked as a `U-node`.

Note also that the computation of the dialectical tree is performed automatically by the DeLP interpreter on the basis of the program available. This process is based on an abstract machine which extends Warren's abstract machine for PROLOG[19]. Given a DeLP program $\mathcal{P}$, solving a query $Q$ with respect to $\mathcal{P}$ may result in four possible answers: YES (there is at least one warranted argument $\mathcal{A}$ for $Q$); NO (there is at least one warranted argument $\mathcal{A}$ for $\sim Q$); UNDECIDED (none of the previous cases hold); and UNKNOWN ($Q$ is not present in the program signature). The emerging semantics is skeptical, computed by DeLP on the basis of the goal-directed construction and marking of dialectical trees, which is performed in a depth-first fashion. Additional facilities (such as visualization of dialectical trees, zoom-in/zoom-out view of arguments, etc.) are integrated in the DeLP environment to facilitate user interaction when solving queries.

## 3. The proposed news recommendation system

The architecture of the proposed news recommender system is presented in figure 1. The system is composed of a News Retriever, which collects news from the Web, social networks and news feeds and stores this data in a database. The database also contains the data associated with the user's trust preferences. A News Manager component is in charge of translating the collected data into the appropriate logical language (e.g. DeLP, A Prolog, etc.) and of generating a list of news backed-up by the user's trust statements. Finally, the News Recommender System component interacts with the end-user, by facilitating the addition of trust preferences and by presenting the final list of recommendations to the user.

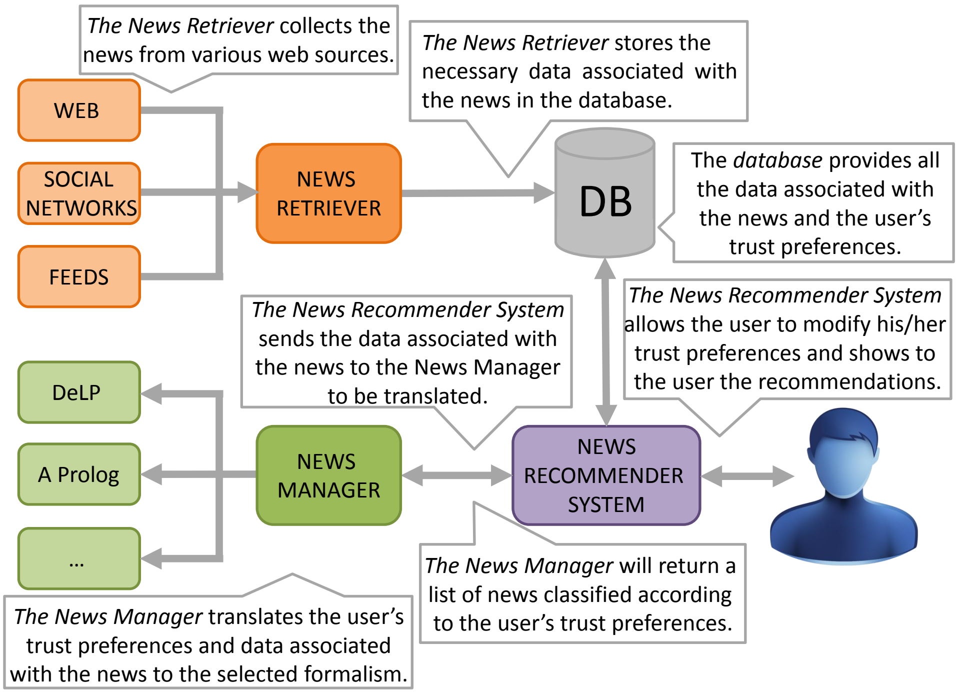


Fig. 1. The proposed trust-based news recommendation system.

The proposed model for trust-based news recommendation takes as a starting point a set of postulates for trust previously reported in [33]. Simply put, our system deals with three different entities: *reports*, *sources* and *viewers*.

- **Reports**. A *news article* or *report* is a written communication of a news event prepared by a specific news agency (source). When a report is made available on the Web, we can identify fields such as *title*, *source*, *timestamp*, *description*, *category* and *link to news content*. Other information related to the report such as *author* can also be derived in certain situations.
- **Sources**. The *source* of a news article is the agency in charge of supplying the report to be used by the media. News can also be published by social networks, web pages or blogs.

- **Viewers**. A *viewer* is a user of the news service. The system maintains a pool of viewers. Viewers can also provide trust statements about reports, sources and other viewers.

We have identified a fundamental relation among these entities, needed to model the concept of trust, which we have called *Trust/Distrust Statements*. A trust (distrust) statement is an explicit assertion of the fact that a viewer trusts (distrusts) a report, a source or another viewer. These statements allow to infer implicit trust relations, which are useful to provide recommendations to the viewer based on trust.

### 3.1. *Using DeLP to model news trust*

We will use the following set of postulates (previously developed in [33]) to model the notion of trust among users, news reports and news sources in an intuitive way.

**Postulate 1.** A report coming from a trusted source will typically be trusted.
**Postulate 2.** A report coming from a distrusted source will typically be distrusted.
**Postulate 3.** A report trusted by a trusted viewer will typically be trusted.
**Postulate 4.** A report distrusted by a trusted viewer will typically be distrusted.
**Postulate 5.** A source trusted by a trusted viewer will typically be trusted.
**Postulate 6.** A source distrusted by a trusted viewer will typically be distrusted.

An interesting situation will arise when two conflicting conclusions can be reached. For example, a trusted viewer distrusts a report, but the report was released by a trusted source. Assuming that in general we prefer to base our opinion on information provided by trusted sources, we can add the following two postulates:

**Postulate 7.** A report coming from a trusted source will typically be trusted, even if it is distrusted by a trusted viewer.
**Postulate 8.** A report coming from a distrusted source will typically be distrusted, even if it is trusted by a trusted viewer.

Another important consideration is that trust is context dependant, which means that we may choose to trust or distrust certain sources or viewers depending on the topic of the report in question. For example, we may trust a news report informing about an accident in our city if the report comes from a local news source but we may not trust the same source reporting on other topics. Similarly, we may trust viewers that have recognized expertise in certain areas but not in others. This can be stated as follows:

**Postulate 9.** A report on a particular topic will typically be trusted if we trust the report's source for that topic.
**Postulate 10.** A report on a particular topic will typically be distrusted if we distrust the report's source for that topic.
**Postulate 11.** A report on a particular topic, which is trusted by another viewer will typically be trusted if we trust that viewer for that topic.

**Postulate 12.** A report on a particular topic, which is distrusted by another viewer will typically be distrusted if we trust that viewer for that topic.

Additional postulates could be added. For example, if some viewer has a very good reputation for fact-checking, we will prefer to trust this viewer's opinion even if it conflicts with other viewers' opinions. Other postulates that could be added to the list may include references to the timestamp of the report. For instance, a more recent report will be trusted over an outdated one, unless the report is just out (due to the eagerness to publish the story some fresh news reports may not be as reliable as old ones). The list of postulates could be extended indefinitely, including references to news author, country of origin of the source, news category, etc. Some postulates could be personalized, because different viewers may disagree on the conclusion that should be adopted given certain facts. For the sake of simplicity, we will take postulates 1 to 12 as the core postulates for our trust-management system.

These postulates can be translated into the following DeLP rules:

$$\begin{array}{rcll}
trust_report(V,R) & \prec & report_source(R,S), trust_source(V,S). & (R_1) \\
{\sim}trust_report(V,R) & \prec & report_source(R,S), {\sim}trust_source(V,S). & (R_2) \\
trust_report(V,R) & \prec & trust_viewer(V,V_1), trust_report(V_1,R). & (R_3) \\
{\sim}trust_report(V,R) & \prec & trust_viewer(V,V_1), {\sim}trust_report(V_1,R). & (R_4) \\
trust_source(V,S) & \prec & trust_viewer(V,V_1), trust_source(V_1,S). & (R_5) \\
{\sim}trust_source(V,S) & \prec & trust_viewer(V,V_1), {\sim}trust_source(V_1,S). & (R_6) \\
trust_report(V,R) & \prec & report_source(R,S), trust_source(V,S), & \\
 & & trust_viewer(V,V_1), {\sim}trust_report(V_1,R). & (R_7) \\
{\sim}trust_report(V,R) & \prec & report_source(R,S), {\sim}trust_source(V,S), & \\
 & & trust_viewer(V,V_1), trust_report(V_1,R). & (R_8) \\
trust_report(V,R) & \prec & is_about(R,T), report_source(R,S), & \\
 & & trust_source_topic(V,S,T). & (R_9) \\
{\sim}trust_report(V,R) & \prec & is_about(R,T), report_source(R,S), & \\
 & & {\sim}trust_source_topic(V,S,T). & (R_{10}) \\
trust_report(V,R) & \prec & is_about(R,T), trust_viewer_topic(V,V_1,T), & \\
 & & trust_report(V_1,R). & (R_{11}) \\
{\sim}trust_report(V,R) & \prec & is_about(R,T), trust_viewer_topic(V,V_1,T), & \\
 & & {\sim}trust_report(V_1,R). & (R_{12})
\end{array}$$

The proposed trust-management system has been tested with built-in rules $R_1$ to $R_{12}$. We should remark that viewers do not need to deal directly with DeLP rules. However, through a user-friendly question-answering interface it will be possible to extend or adjust the built-in core rules based on the viewer's preferences. Trust and distrust statements about reports, sources and other viewers will be added to the system whenever the viewer rates these entities.

For a particular viewer $v$, and based on the corresponding DeLP rules and facts, news reports will be classified into four sets:

**Trusted Reports:** those reports $r_i$ for which there exists at least one warranted argument supporting $trust\_report(v, r_i)$.

**Distrusted Reports:** those reports $r_i$ such that there is a warranted argument supporting $\sim trust\_report(v, r_i)$.

**Undecided:** those reports $r_i$ for which there is no warranted argument for $trust\_report(v, r_i)$ or $\sim trust\_report(v, r_i)$.

**Unknown:** those reports $r_i$ for which there is no information for $trust\_report(v, r_i)$ or $\sim trust\_report(v, r_i)$.

This classification will allow the viewer to focus on those reports considered trustworthy, and to be warned about the non trustworthy ones.

## 4. On the suitability of DeLP to model Trust

As has been pointed out by previous studies, no objective basis of trust exists [8]. Although this precludes an empirical evaluation of our model it is possible to analyze its suitability by means of a set of representative examples and a theoretical analysis of its mains characteristics. Therefore, the goal of this section is to provide evidence to support the aptness of DeLP as a mechanism for modeling the notion of trust. We do so by presenting a set of selected examples that illustrate the operation of the proposed approach and by performing a theoretical analysis of some of its properties.

### 4.1. *Some selected examples*

In order to determine if a report is trusted, distrusted or undecided the system takes as a basis the information stored in the database about users, news sources, news reports and trust judgments. This information is translated into DeLP facts, which are added to the proposed postulates to complete the trust model. Once this is done, the service is able to automatically determine which reports can be trusted by a particular user.

Suppose that Ana is a user who has logged into the system and has already expressed her trust judgments about other users and various news sources and reports. Let "google_hits_one_billion" be a news report informing that Google received 1 billion of unique visitors during May 2011. Suppose that after being translated into DeLP facts, the system's trust information is represented by the following facts:

$$
\begin{array}{l}
report\_source(google\_hits\_one\_billion, slashdot) \\
report\_source(facebook\_hits\_one\_billion, etc\_news) \\
report\_source(microsoft\_hits\_one\_billion, msn\_news) \\
trust\_source(ana, slashdot) \\
\sim trust\_source(cristian, etc\_news) \\
\sim trust\_report(marcela, google\_hits\_one\_billion) \\
trust\_report(marcela, facebook\_hits\_one\_billion) \\
\sim trust\_report(marcela, microsoft\_hits\_one\_billion) \\
trust\_report(cristian, microsoft\_hits\_one\_billion)
\end{array}
$$

$$trust\_viewer(ana, emanuel)$$
$$trust\_viewer(ana, cristian)$$
$$trust\_viewer(emanuel, marcela)$$

Based on this and the postulates described earlier, the system is able to classify reports as trusted, distrusted, undecided or unknown and use this information at the moment of presenting suggestions to the user. In this case, suppose that the system needs to classify the reports "google_hits_one_billion", "facebook_hits_one_billion" and "microsoft_hits_one_billion".

In order to decide how to classify each report, the system internally tries to find a warranted argument associated with trust judgments about each of them. In figure 2 we can see that there is a warranted argument supporting the statement $trust\_report(ana, google\_hits\_one\_billion)$, so the report should be trusted by Ana. In figure 3 we can see the existence of a warranted argument for $\sim trust\_report(ana, facebook\_hits\_one\_billion)$, so this report should be distrusted by Ana. Finally, figure 4 shows that the report cannot be trusted, because there is not a warranted argument for $trust\_report(ana, microsoft\_hits\_one\_billion)$. In addition, the system cannot conclude that Ana should distrust this report because it is not possible to find a warranted argument for $\sim trust\_report(ana, microsoft\_hits\_one\_billion)$. Therefore, the trust status of this report stands as undecided.

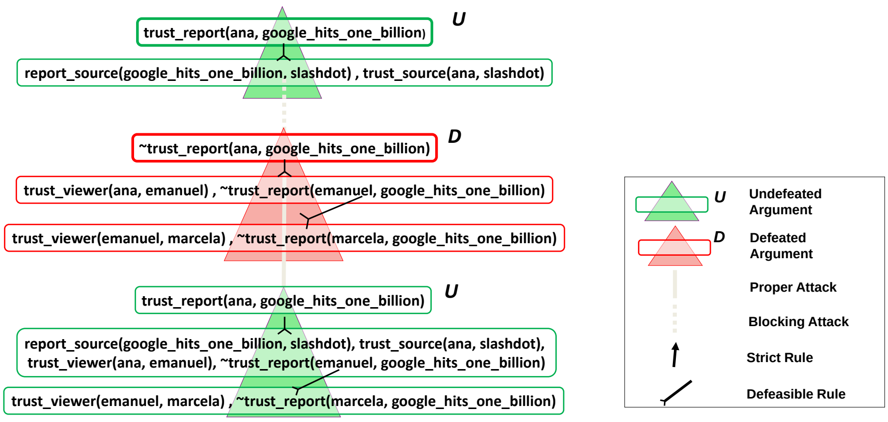


Fig. 2. DeLP dialectical tree showing the reasons to trust google_hits_one_billion.

With these examples we can see how trust is spread among users and what possible scenarios are possible for the epistemic status of a user regarding a specific report.

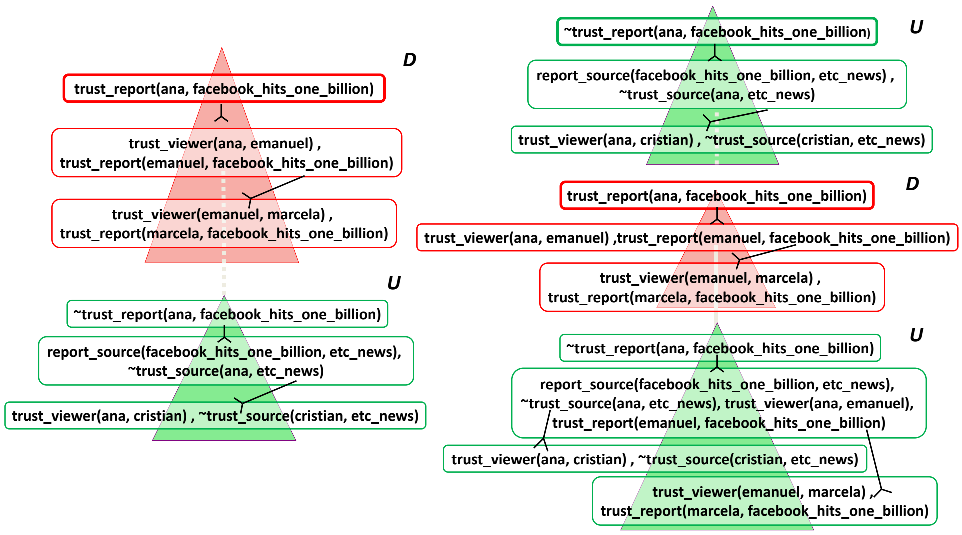


Fig. 3. DeLP dialectical tree showing the reasons to distrust facebook_hits_one_billion.

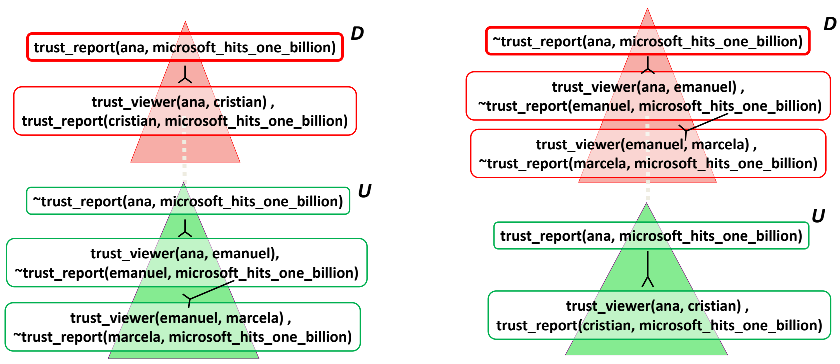


Fig. 4. DeLP dialectical tree showing no reasons to trust or distrust microsoft_hits_one_billion.

### 4.2. *Properties of trust and their characterization using DeLP*

A suitable model of the notion of trust must exhibit some desirable properties. We analyze these properties, showing that DeLP provides a natural framework to characterize and support them.

#### 4.2.1. *Trust is subjective*

Like many aspects of the Web, news is becoming a collaborative activity. However, judgment of news credibility is idiosyncratic. Therefore, models that deal with trust as an objective notion are unrealistic. In this case, the opinion from the“wisdom of the crowds" may not be as useful as the viewer's personal opinion, or the opinion of another trusted viewer. In our working example, we can identify three different epistemic status classes associated with users Ana, Cristian and Marcela regarding the credibility of the report "microsoft_hits_one_billion":

- The answer for $trust\_report(ana, microsoft\_hits\_one\_billion)$ is *undecided*.
- The answer for $trust\_report(cristian, microsoft\_hits\_one\_billion)$ is *yes*.
- The answer for $trust\_report(marcela, microsoft\_hits\_one\_billion)$ is *no*.

#### 4.2.2. *Trust is dynamic and defeasible*

News agencies are subject to time constraints, which results in the publication of reports with incomplete or inaccurate information. Therefore, trust on such reports could be revoked by the release of other more recent ones. In the meantime, news reports trusted by some viewer can be superseded by other reports carrying more authority (for example, coming from some other trusted source). DeLP can effectively deal with the defeasible nature of trust. In our example, we have seen that the answer for $trust\_report(ana, facebook\_hits\_one\_billion)$ is *no*. However, if we add the fact $trust\_source(ana, etc\_news)$, the answer becomes *yes*. On the other hand, the answer for $trust\_report(emanuel, facebook\_hits\_one\_billion)$ is *yes*, but becomes negative if we add the fact $\sim trust\_source(emanuel, etc\_news)$. Although we do not explicitly model the notion of time, our system captures the dynamicity of trust by generating updated conclusion that might be different from those obtained before adding new facts.

#### 4.2.3. *Trust is not always symmetrical*

A viewer may trust another viewer, while a trust relation from the second to the first viewer may be absent. In our example, we can observe that $trust\_viewer(emanuel, marcela)$ holds, but $trust\_viewer(marcela, emanuel)$ is undecided.

#### 4.2.4. *Trust is not always transitive*

Trust does not propagate in an unconstrained manner and therefore is not transitive. Note in our working example that in spite of the existence of facts such as $trust\_viewer(ana, emanuel)$, $trust\_viewer(emanuel, marcela)$, and $trust\_report(marcela, facebook\_hits\_one\_billion)$, the answer obtained for $trust\_report(ana, facebook\_hits\_one\_billion)$ is *no*.

#### 4.2.5. *Trust is context dependant*

As mentioned earlier, a viewer may choose to trust or distrust certain sources or viewers depending on the topic of the report in question. Suppose that our working example is extended with the following facts:

$report\_source(vicodin\_relieves\_moderate\_to\_severe\_pain, medical\_news).$
$trust\_viewer\_topic(cristian, house, medicine).$
$is\_about(vicodin\_relieves\_moderate\_to\_severe\_pain, medicine).$
$trust\_report(house, vicodin\_relieves\_moderate\_to\_severe\_pain).$

The above facts indicate that Cristian trusts House's judgment to decide whether a news report on Medicine is credible. In this case, since House trusts the medical news report "vicodin_relieves_moderate_to_severe_pain", the answer obtained for $trust\_report(cristian, vicodin\_relieves\_moderate\_to\_severe\_pain)$ will be *yes*.

On the other hand, assume that House has also indicated that he does not trust the report "barça_is_the_best_football_team_of_all_time". This report appears in "sports_news", a news source trusted by Cristian. This is represented by the following facts:

$\sim trust\_report(house, barça\_is\_the\_best\_football\_team\_of\_all\_time).$
$report\_source(barça\_is\_the\_best\_football\_team\_of\_all\_time, sports\_news).$
$trust\_source(cristian, sports\_news).$

In this case, in spite of House's judgment regarding the credibility of this report, the answer to $trust\_report(cristian, barça\_is\_the\_best\_football\_team\_of\_all\_time)$ will be *yes*.

### 4.3. *A theoretical analysis*

Our recommendation system is currently based on an argumentative framework, the DeLP system. Argumentation logics, and non-monotonic logics in general, do not exhibit the usual properties of classical theories, but are preferred to these logics for their expressive power and flexibility. To characterize non-monotonic logics Dov Gabbay [18] pioneered the comparison of non-monotonic theories with respect to a set of desirable properties. He endorsed focusing our attention on the properties of the inference relation induced by the formal theory, that is, the relation between conclusions and the set of premises supporting them. Further pursuing this approach, first Kraus *et al.* [24], and later Makinson [26], studied the set of core properties every non-monotonic theory ought to have. *Inclusion*, *idempotence* and *cautious monotonicity* were identified as a set of core properties that non-monotonic theories should verify. Systems that verify cut and cautious monotonicity are said to be *cumulative*. But trying to characterize argumentation theories with these properties did not lead to encouraging results [34,32]. Unfortunately, there is not yet consensus in the argumentation community on a given set of properties that argumentative inference should comply with.

In this work, to formally analyze a system based on argumentation we have chosen to consider two simple but important properties. First we consider *inclusion*, which is clearly desirable in the context of any sensible inference relation, as it stands for accepting those premises upon which we reason. In what follows we analyze the inclusion property in the context of our recommender system.

**Proposition 4.1.** *Consider a recommender system using a DeLP program $\mathcal{P} = (\Pi, \Delta)$ to derive its conclusions. Let $\mid\!\sim$ denote the DeLP inference operator, that allows to derive a given warranted conclusion from a DeLP program. Let $C(\Pi) = \{\phi \mid \Pi \mid\!\sim \phi\}$ be a consequence that denotes the set of warranted conclusions that can be obtained from the set of facts $\Pi$ using the program $\mathcal{P}$ then $\Pi \subseteq C(\Pi)$*

**Proof.** Let $\phi$ be a fact in $\Pi$. Thus $\phi$ is supported by an empty argument. Empty arguments cannot be attacked nor defeated by an argument, and then $\phi$ is a warranted conclusion from the program $\mathcal{P}$, that is, $\phi \in C(\Pi)$. □

This may seem a trivial result, but some argumentation theories that have been used to implement systems similar to ours (such as the ASPIC system[11]) do not satisfy this property [34].

Another property that is reasonable for any inference mechanism is *consistency*. Consistency is a basic property for the news recommender system, in the sense that it is not reasonable that for a given user $v$ the system simultaneously classifies a given report as trusted and distrusted. In our recommended system it is not possible to conclude for a user $v$ and a report $r_i$ $trust\_report(v, r_i)$ and $\sim trust\_report(v, r_i)$. In order to show that this is the case, we will prove that the set of arguments that can be warranted from a given DeLP program are not conflicting among themselves. To show this property we will first demonstrate some auxiliary results.

**Proposition 4.2.** *Let $\mathcal{P} = (\Pi, \Delta)$ be a program and $\langle \mathcal{A}, Q\rangle$, $\langle \mathcal{B}, R\rangle$ two arguments built from $\mathcal{P}$such that $\langle \mathcal{A}, Q\rangle$ is a defeater of $\langle \mathcal{B}, R\rangle$ and $\langle \mathcal{A}, Q\rangle$ is warranted wrt $\mathcal{P}$. Let $\langle \mathcal{A}_s, Q_s\rangle$ be a supporting argument of $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$ such that $\langle \mathcal{A}_s, Q_s\rangle$ is a `U node`. If $\langle \mathcal{A}_s, Q_s\rangle$ appears in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$ it must be also labeled as a `U node` in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$.*

**Proof.** To show this property we use structural induction on the subtree rooted in $\langle \mathcal{A}_s, Q_s\rangle$.

**Base case** Let us consider the fact that $\langle \mathcal{A}_s, Q_s\rangle$ does not have defeaters in $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$. If $\mathcal{A}_s$ belongs to $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$ then it must be marked as a `U node`, given that no new defeaters can be built.

**Inductive step** Let us assume that $\langle \mathcal{A}_s, Q_s\rangle$ has $k$ defeaters in $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$, noted as $\mathcal{B}_1, \mathcal{B}_2 \ldots, \mathcal{B}_k$, and $\langle \mathcal{A}_s, Q_s\rangle$ is present in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$. Since $\mathcal{A}_s$ is a `U node` in $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$ every defeater of $\mathcal{B}_i$ must be a `D node` in $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$.

Next we will show that if $\mathcal{B}_i$ is present in $\mathcal{T}_{\langle \mathcal{B}, P\rangle}$ then it must be labeled as a `D node` in this tree. Given that $\mathcal{B}_i$ is a `D node` in $\mathcal{T}_{\langle \mathcal{A}, Q\rangle}$ there must exists a non-defeated supporting argument $\mathcal{C}$ in the subtree rooted in $\mathcal{B}_i$.

If we use the inductive hypothesis on this subtree it follows that if $\mathcal{C}$ is present in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$ it must be labeled as a *U node* and then $\mathcal{B}_i$ must be a `D node` in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$. It remains to check what happens when $\mathcal{C}$ cannot be introduced as a defeater of $\mathcal{B}_i$ in $\mathcal{T}_{\langle \mathcal{B}, R\rangle}$. This may follow from one of these reasons:

(1) $\mathcal{C}$ is contradictory with an interfering argument in the argumentation line in which $\mathcal{C}$ should be added.
(2) $\mathcal{C}$ is a sub-argument of a previous argument in this argumentation line.

The first scenario is not possible, given that in this case $\mathcal{C}$ could not appear in $\mathcal{T}_{\langle \mathcal{A},Q\rangle}$ since it would be contradictory with a supporting argument in the argumentation line. In the second case, $\mathcal{C}$ must be a sub-argument of $\mathcal{B}$ (otherwise it could not belong to $\mathcal{T}_{\langle \mathcal{A},Q\rangle}$). Nevertheless, since $\mathcal{C}$ is a defeater of $\mathcal{B}_i$, the set $\mathcal{C} \cup \mathcal{B}_i \cup \Psi$ allows the derivation of complementary literals. Then $\mathcal{B} \cup \mathcal{B}_i \cup \Psi$ also allows the derivation of complementary literals since $\mathcal{C} \subseteq \mathcal{B}$. In this situation it is not possible that $\mathcal{B}_i$ be in $\mathcal{T}_{\langle \mathcal{B},R\rangle}$, given that it contradicts a supporting argument in its argumentative line. Thus, if $\mathcal{B}_i$ is in $\mathcal{T}_{\langle \mathcal{B},R\rangle}$ it must be labeled as a `U node` in this tree. □

The following statement derives from the previous proposition.

**Corollary 4.1.** *Let $\mathcal{P}$ be a program and $\langle \mathcal{A}, Q\rangle$, $\langle \mathcal{B}, R\rangle$ two arguments built from $\mathcal{P}$. If $\langle \mathcal{A}, Q\rangle$ is a defeater of $\langle \mathcal{B}, R\rangle$ and $\langle \mathcal{A}, Q\rangle$ is a warranted argument wrt $\mathcal{P}$ then $\langle \mathcal{A}, Q\rangle$ must be labeled as a `U node` in the dialectical tree rooted in $\langle \mathcal{B}, R\rangle$.*

**Proof.** By hypothesis, argument $\langle \mathcal{A}, Q\rangle$ is a supporting undefeated argument in $\mathcal{T}_{\langle \mathcal{A},Q\rangle}$ that is also present in $\mathcal{T}_{\langle \mathcal{B},R\rangle}$. From proposition 4.2 we can infer that $\langle \mathcal{A}, Q\rangle$ must be a `U node` in the dialectical tree for $\langle \mathcal{B}, R\rangle$. □

Finally, the following lemma proves consistency of the recommendations.

**Lemma 4.1.** *Consider a user $v$, a report $r_i$, and a program $\mathcal{P} = (\Pi, \Delta)$ that contains the DeLP facts and rules. Then it is not possible to conclude from $r_i$ $trust\_report(v, r_i)$ and $\sim trust\_report(v, r_i)$.*

**Proof.** Suppose by contradiction that there exists $v$ and $r_i$ such that $trust\_report(v, r_i)$ and $\sim trust\_report(v, r_i)$ can be derived in the system. Then there exists a warranted argument $\mathcal{A}$ that supports $trust\_report(v, r_i)$ and a warranted argument $\mathcal{B}$ that supports $\sim trust\_report(v, r_i)$. Let $\mathtt{Warr}(\mathcal{P})$ be the set of warranted arguments in $\mathcal{P}$. Both $\mathcal{A}$ and $\mathcal{B}$ are should be in $\mathtt{Warr}(\mathcal{P})$. In this case $\mathcal{A}$ attacks $\mathcal{B}$

We can assume without any loss of generality that $\mathcal{A}$ defeats $\mathcal{B}$. Since $\mathcal{A}$ and $\mathcal{B}$ are warranted we can assume that there exists a dialectical tree with root in $\mathcal{A}$, noted $\mathcal{T}_{\langle \mathcal{A}, trust\_report(v,r_i)\rangle}$, that warrants $trust\_report(v, r_i)$ and a dialectical tree $\mathcal{T}_{\langle \mathcal{B}, \sim trust\_report(v,r_i)\rangle}$ that warrants $\sim trust\_report(v, r_i)$.

Argument $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$ defeats $\langle \mathcal{B}, \sim trust\_report(v, r_i)\rangle$ and thus $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$ must be marked as a `D node` in the dialectical tree $\mathcal{T}_{\langle \mathcal{B}, \sim trust\_report(v,r_i)\rangle}$, since otherwise $\langle \mathcal{B}, \sim trust\_report(v, r_i)\rangle$ could not be a warranted argument. However, applying corollary 4.1 argument $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$

must be labeled as a `U node` in the dialectical tree for $\langle \mathcal{B}, \sim trust\_report(v, r_i)\rangle$ (since $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$ is warranted and $\langle \mathcal{B}, \sim trust\_report(v, r_i)\rangle$ defeats $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$). This contradiction arises from assuming the existence of a pair of arguments $\langle \mathcal{A}, trust\_report(v, r_i)\rangle$, $\langle \mathcal{B}, \sim trust\_report(v, r_i)\rangle$ under the conditions previously stated. $\square$

To sum up, argumentation theories are still waiting for the definition of a set of properties that characterize them and can be used as a benchmark for these theories. Meanwhile, we endorse that inclusion and consistency are two basic properties for any argumentation theory, and should be considered as a starting point in any formal analysis. In particular, we contend they are desirable properties for any system used to model the notion of trust.

## 5. Validation by simulation

In order to validate our system, we ran eight simulations using synthetic data. Figure 5 outlines the architecture instance that have been implemented to conduct the simulations.

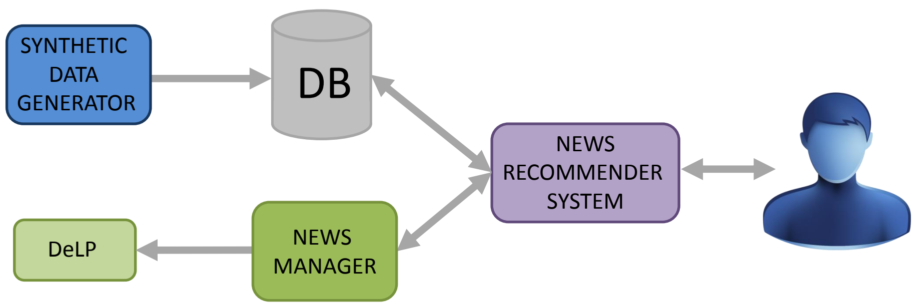


Fig. 5. The architecture instance implemented to run the simulations.

In each of the performed simulations, we generated a different number of synthetic users and reports to analyze how the system behaved. Another value that varied from one simulation to another was the interaction between the systems elements. By interaction we mean the expression of trust and distrust regarding reports, sources and users by each of the users. Figure 6 summarizes the setting data of each simulation run and the execution times on a AMD Phenom II processor with 6 cores and 8 GB RAM.

It is important to emphasize how these simulations allowed us to further validate the proposed framework. Figure 7 presents the initial trust relations between 20 users and 50 reports. After running Simulation 7 the system converges to the trust relations shown in figure 8. This figure provides good evidence of the inference power

| | #Users | #Reports | Interaction | Time |
|---|---|---|---|---|
| Sim 1 | 5 | 4 | 5% | 4 sec |
| Sim 2 | 5 | 4 | 25% | 6 sec |
| Sim 3 | 5 | 4 | 40% | 3 min 57 sec |
| Sim 4 | 10 | 9 | 5% | 19 sec |
| Sim 5 | 10 | 9 | 25% | 50 min 23 |
| Sim 6 | 15 | 12 | 25% | 1 hour 32 min 16 sec |
| Sim 7 | 20 | 50 | 10% | 4 min 8 sec |
| Sim 8 | 20 | 50 | 50% | 5 hours 29 min 47 sec |

Fig. 6. Simulation setting and execution times.

of the framework. By significantly reducing the number of "undecided" entries we can gain confidence on the recommendations that can be presented to the user. In addition, we can visually validate two important properties discussed earlier: *inclusion*, i.e., the explicit viewer's credibility status regarding a news report cannot be revoked, and *consistency*, i.e., the system cannot conclude that an entity is simultaneously trusted and distrusted by the same viewer.

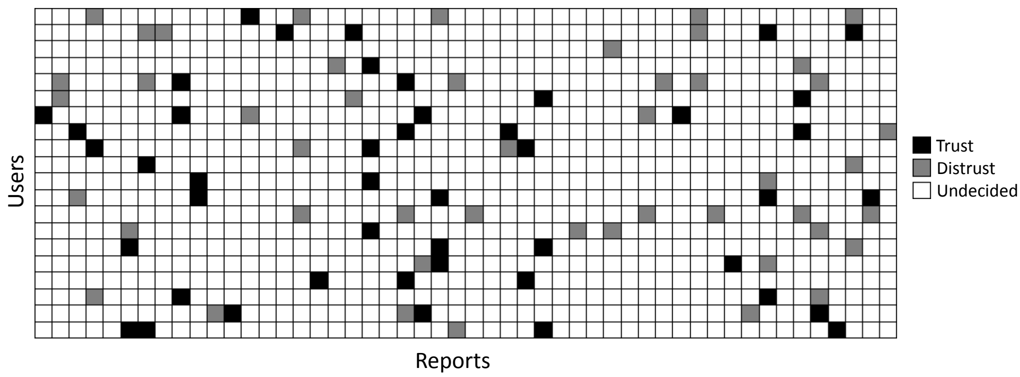


Fig. 7. Explicit trust relations for Simulation 7.

## 6. Technical issues

Based on the model presented before, we have implemented a novel system for news recommendations, based on the architecture described next. We have chosen to use a client-server style architecture, given that it can accommodate our requirements in a natural way.

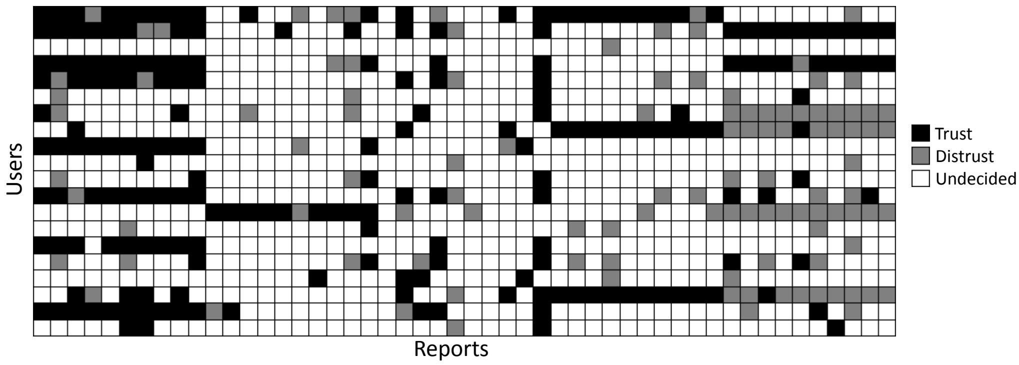


Fig. 8. Explicit and inferred trust relations for Simulation 7.

At this point, we have developed two different types of services. The first one provides news that were obtained from the available sources. The second one deals with news trust. This service must be able to decide if a given news report could be trustworthy for the currently logged in user. So far, we have focused on the implementation of the second service, using a DeLP interpreter. Note that even though the trust manager is currently implemented using DeLP, the system is designed to allow a seamless transition to a different system, given that it was built taking into account modularity as a fundamental design principle.

The recommender system was implemented as a web application, because web applications can be easily accessed from any computer or location equipped with a browser and Internet access (accessibility and portability). Most of the work is done on the server, resulting in low resource consumption and giving the application independence from the underlying operating system to avoid compatibility problems (efficiency and multi-platform). In addition, the ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for the popularity of web based applications. This was another factor that influenced the selection of DeLP as the underlying reasoning system, since it is implemented as a web service.

The language chosen for development is PHP [5], which is a relatively new language designed for the sole purpose of creating web applications. This means that the most common tasks in developing these applications can be easily, quickly and effectively accomplished by using PHP. In addition, it is a multi-platform and non-proprietary language. A normal PHP script can be executed without changing a single line of code on any server with a PHP interpreter, i.e. Windows, Linux etc. A prototype of the developed system can be downloaded at http://code.google.com/p/newsrecomender/. In addition, its basic functionality can be tested at http://newsrecommender.com.ar.

A main concern with the model proposed here certainly is its time complexity. Complexity of the DeLP systems was first analyzed in [15] where in the worst case inference was shown to be in the order of $O\left(2^{|\Delta|^{3.(2^{|\Delta|})/4}}\right)$. Later, new algorithms for the inference process of DeLP were developed in [12], that relied on the use of precompiled knowledge to speed-up inference. Using these algorithms complexity is lowered to $O(2^{|\Delta|}.|\Delta|)$ [13]. This result could still be problematic when the number of rules in $\Delta$ can be arbitrarily large, but this is not the case for our model since the number of rules is fixed and typically small. Even though the theoretical results regarding the system complexity in the worst case are not so encouraging, our simulations have indicated that for realistic situations the system performance is acceptable, as shown in figure 6.

Another issue that needs to be addressed is the "cold start problem", which is prevalent in recommender systems. This problem typically affects those users that have not provided explicit trust statements to the system. To overcome this problem we propose a simple solution, which consists in requiring that each new user enters at least one trust statement, making it explicit that he or she trusts some other user or a source that is already represented in the system.

A notion related to the concept of trust is "reputation". Reputation is typically used to facilitate trust and is commonly represented using scores. Differently from trust, which is subjective and represented as a relation between a user and an entity, reputation is an objective value. Although our system in its current stage is not intended to compute reputation scores, we expect to address this issue in the future. In particular, we plan to look into two questions: (1) how to use explicit and implicit trust relations to compute reputation scores, and (2) how to take advantage of reputation scores in the absence of trust statements for a particular user or item, to alleviate the cold start problem. In order to implement a trustworthy system we will need to deal with unfair ratings from dishonest raters. This will be part of our future work.

## 7. Future work

Although we have analyzed a representative set of postulates for modeling the notion of trust, we are aware that there are many other potentially useful postulates that a user might want to add to the system. In particular, a user might want to include postulates that give preference to fresh news over old ones. At this point such postulates would need to be hand-coded as DeLP rules. In the future we expect to build a more user-friendly interface to facilitate entering this information directly by the end user without the need to edit the postulates as DeLP rules. This information will then be automatically translated to DeLP rules.

In the future, we expect to test the system using other logics and interpreters, as well as with other mechanisms for trust propagation. It is worth mentioning that if we change the underlying logic, the system may not exhibit the desired behavior for trust propagation, which is guaranteed by DeLP. In addition, we plan to integrate the tool with social networks' APIs, such as the ones provided by Facebook or

Twitter, to collect large amounts of trust statements about news reports, news sources and between users. The proposed trust propagation mechanism could be naturally integrated with a distributed system, to share not only news but also other kinds of information items. In this sense, every user will be able to contribute to and collaborate with specific communities as well as the full network of users.

Another promising future research direction consists in combining explicit trust/distrust statements with cumulative numerical data. To do so we plan to use a different argumentation system where quantitative arguments could be constructed. Another interesting future research avenue will consist in implementing machine learning mechanisms that learn new trust statements by monitoring the user's behavior. The resulting model will combine the benefits of having explicit statements provided by the user and new statements generated and updated automatically as the system learns from the user behavior.

## 8. Conclusions

We have presented a theoretical framework that can be applied to implement a trust-based recommendation system for news on the Web. The first main contribution of this work is the analysis of a set of properties of trust, such as being subjective, dynamic, defeasible, context dependant and not always symmetrical or transitive. We show that these properties can all be naturally modeled using the proposed framework.

Our second main contribution is the proof of two fundamental theorems that characterize the behavior of any recommender system based on the proposed model of trust propagation. The first theorem establishes that an explicit trust premise cannot be revoked by the system. More formally, it states that the set of trust premises are included in the set of trust conclusions. The second theorem establishes the consistency of the system, which precludes the possibility of concluding that a viewer simultaneously trust and distrust the same entity.

This proposal differs from previous work on news recommendation systems in allowing to draw logical conclusions about the credibility of news reports based on the opinions of a set of users. At the present time, the news management system has been implemented and tested with a set of synthetic users, and has demonstrated a natural and convenient treatment of the defeasible nature of trust.

We expect to release a beta version of the system in the near future. This will allow us to empirically evaluate our proposal. We anticipate that a large number of trust statements, combined with the great variety of news services currently available, will reveal the great potential of qualitative approaches to news recommendation.

## References

1. http://news.google.com/. Accessed Feb. 2013.
2. http://newstrust.net/. Accessed Feb. 2013.

3. http://news.yahoo.com/. Accessed Feb. 2013.
4. http://www.msnbc.msn.com/. Accessed Feb. 2013.
5. http://www.php.net/. Accessed Feb. 2013.
6. Banos, E., Katakis, I., Bassiliades, N., Tsoumakas, G., Vlahavas, I., *PersoNews: A Personalized News Reader Enhanced by Machine Learning and Semantic Filtering*, 5th International Conference on Ontologies, DataBases, and Applications of Semantics (2006).
7. Bedi, P., Kaur, H., *Trust based personalized recommender system.* INFOCOM Journal of Computer Science. (2006).
8. Benedicktus, R.L., *The effects of 3rd party consensus information on service expectations and online trust.* Journal of Business Research 64(8). (2011), pp. 846–853.
9. Bikakis,A., Antoniou, G. *Defeasible Contextual Reasoning with Arguments in Ambient Intelligence.* IEEE Trans. Knowl. Data Eng. 22(11): 1492-1506 (2010)
10. Brin, S., Page, L., *The anatomy of a large-scale hypertextual Web search engine.* Computer Networks and ISDN Systems 30(1–7). (1998), pp. 107–117.
11. Caminada, M., Amgoud, L., *On the evaluation of argumentation formalisms.* Artificial Intelligence 171(5–6). (2007), pp. 286–310.
12. Capobianco, M., Chesñevar, C.I., Simari, G.R., *Argumentation and the dynamics of warranted beliefs in changing environments.* Journal of Autonomous Agents and Multiagent Systems 11. (2005), pp. 127–151.
13. Capobianco, M., Simari, G., *A proposal for making argumentation computationally capable of handling large repositories of uncertain data.* In: Proceedings of the third international conference on scalable uncertainty management. (2009)
14. Carbone, M., Nielsen, M., Sassone, V. *A Formal Model for Trust in Dynamic Networks.* In: Proc. of International Conference on Software Engineering and Formal Methods. (2003), pp. 54–63.
15. Cecchi, L., Fillottrani, P., Simari, G., *On the complexity of delp through game semantics.* In: NMR 2006, 11th International Workshop on Nonmonotonic Reasoning. (2006)
16. Chesñevar, C., Maguitman, A., Gonzlez, M.P., *Empowering recommendation technologies through argumentation.* In: Argumentation in Artificial Intelligence. (Springer Verlag, 2010).
17. Czenko, M., Doumen, J., Etalle, S., *Trust management in p2p systems using standard tulip.* (2008)
18. Gabbay, D., *Theoretical foundations for nonmonotonic reasoning in expert systems.* In: Apt, K. (ed.) Logics and Models of Concurrent Systems. (Springer-Verlag, Berlin, Germany, 1985)
19. García, A., Simari, G., *Defeasible logic programming: An argumentative approach.* Theory and Practice of Logic Programming 4(1). (2004), pp. 95–138.
20. Guha, R., Kumar, R., Raghavan, P., Tomkins, A., *Propagation of trust and distrust.* In: International World Wide Web Conference. (2004).
21. Hess, C., *Trust-based recommendations for publications: A multi-layer network approach.* TCDL Bulletin 2(2). (2006).
22. Huang, J., Fox, M.S., *An ontology of trust: formal semantics and transitivity.* In: Proceedings of the 8th international conference on Electronic commerce: The new e-commerce: innovations for conquering current barriers, obstacles and limitations to conducting successful business on the internet. (ACM, New York, USA, 2006), pp. 259–270.
23. Jamali, M., Ester, M., *A matrix factorization technique with trust propagation for recommendation in social networks.* In: Proceedings of the fourth ACM conference on

Recommender systems. (ACM, New York, USA, 2010), pp. 135–142.
24. Kraus, S., Lehmann, D., Magidor, M., *Non-monotonic reasoning, preferential models and cumulative logics.* Artificial Intelligence 44. (1990), pp. 167–207.
25. Kravari, K., Bassiliades, N,., *HARM: A hybrid Rule-Based agent reputation model based on temporal defeasible logic rules on the web: Research and applications.* volume 7438 of Lecture Notes in Computer Science, chapter 15, (Springer Berlin, 2012).
26. Makinson, D., *General patterns in nonmonotonic reasoning.* In: Gabbay, H., Robinson (eds.) Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3. (Oxford University Press, 1994), pp. 35–110.
27. Muller, T., *Semantics of trust.* In: Proceedings of the 7th International conference on Formal aspects of security and trust. (Springer-Verlag, Berlin, Heidelberg, 2011), pp. 141–156.
28. Nagura, R., Seki, Y., Kando, N., Aono, M., *A method of rating the credibility of news documents on the web.* In: SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. (ACM Press, New York, USA, 2006), pp. 683–684.
29. Ofuonye, E., Beatty, P., Reay, I., Dick, S., Miller, J., *How do we build trust into e-commerce web sites?* IEEE Software 25. (2008), pp. 7–9.
30. Park, S., Kang, S., Song, J., *Polynews: Delivering multiple aspects of news to mitigate media bias.* Technical report, KAIST. (2006).
31. Parsons, S., McBurney, P., Sklar, E. *Reasoning about trust using argumentation: A position paper.* In: Proceedings of the 7th Workshop on Argumentation in Multiagent Systems (2010).
32. Prakken, H., Vreeswijk, G., *Logics for defeasible argumentation.* In: Gabbay, D., Guenthner, F. (eds.) Handbook of Philosophical Logic, vol. 4. (Kluwer Academic Publishers, 2002), pp. 218–319.
33. Sagui, F.M., Maguitman, A.G., Chesñevar, C.I., Simari, G.R., *Modeling news trust: A defeasible logic programming approach.* Iberoamerican Journal of Artificial Intelligence 12(40). (2009), pp. 63–72.
34. Stankevicius, A.G., Capobianco, M., *On cummulativity in the context of defeasible argumentation.* In: XIV Congreso Argentino de Ciencias de la Computacin, CACIC 2008 (2008).
35. Villata, S., Boella, G., Gabbay, D.M., Van Der Torre, L., *Arguing about the trustworthiness of the information sources.* In: Proceedings of the 11th European conference on Symbolic and quantitative approaches to reasoning with uncertainty. (Springer-Verlag, Berlin, Heidelberg, 2011), pp. 74–85.
36. Walter, F., Battiston, S., Schweitzer, F., *A model of a trust-based recommendation system on a social network.* Autonomous Agents and Multi-Agent Systems 16. (2008), pp. 57–74.