APOPSIS : A Web-based Platform for the Analysis of Structured Dialogues

Elisjana Ymeralli

Thesis submitted in partial fulfillment of the requirements for the

*Masters’ of Science degree in Computer Science*

University of Crete
School of Sciences and Engineering
Computer Science Department
Voutes University Campus, Heraklion, GR-70013, Greece

Thesis Advisor: Prof. Dimitris Plexousakis

This work has been performed at the University of Crete, School of Sciences and Engineering, Computer Science Department.

The work has been supported by the Foundation for Research and Technology - Hellas (FORTH), Institute of Computer Science (ICS).
APOPSIS : A Web-based Platform for the Analysis of Structured Dialogues

Ymeralli Elisjana

Copyright © by Ymeralli Elisjana, 2016
All Rights Reserved
University of Crete
Computer Science Department

APOPSIS : A Web-based Platform for the Analysis of Structured Dialogues

Thesis submitted by
Elisjana Ymeralli
in partial fulfillment of the requirements for the Masters’ of Science degree in Computer Science

THESIS APPROVAL

Author: 
Elisjana Ymeralli

Committee approvals:
Dimitris Plexousakis
Professor, University of Crete
Thesis Supervisor

Yannis Tzitzikas
Assistant Professor, University of Crete
Committee Member

Giorgos Flouris
Researcher, Committee Member

Departmental approval:
Antonis Argyros
Professor, University of Crete
Director of Graduate Studies

Heraklion, December 2016
APOPSIS : A Web-based Platform for the Analysis of Structured Dialogues

Abstract

Social networks are constantly evolving to support the increasing needs for knowledge sharing, interaction and collaboration among people from all over the world through Web. These networks provide to users many interaction platforms where they can share their opinions and their life experiences. Debate portals are one type of such platforms where people can express their views in the form of arguments and participate in support of or against issues occurring in the dialogue in a more structured way. As long as social networks and other sources of discussion produce more and more content, the need to understand and summarize the opinions expressed within dialogues increases, in order to reduce the burden of having to go through the entire debate.

The system provides a debating environment that aims to motivate people to participate in structured, goal-oriented dialogues. As a debating platform, it enables users to raise issues, ask their own questions, post supporting or counter-arguments, comment and vote. The overall objective is to offer different means of analysis of the debates, in order for the participants to obtain a complete picture of the validity and justification strength of each individual opinion expressed, as well as of the acceptance of the positions issued within each debate. The system provides a range of functionalities, the most important of which concern the creation of new topics of discussion, the evaluation of arguments with different metrics, and the analysis of various aspects of the dialogues.

In this work, we start with providing an argument map for modeling discussions and the relations between them with debate elements such as issue, position, pro-argument and con-argument. Then, we apply an existing formal framework for evaluating the strength of arguments, called sm-Dice. Every argument strength is calculated based on a multi-aspect evaluation. Next, we implement a debate analysis by taking into account the various aspects of the dialogues. This analysis covers different information needs emerging from users, in order to summarize various aspects of a debate, focusing not only on arguments, but also on user profile characteristics throughout the decision-making process. A collection of machine learning algorithms is applied for the clustering of features and the extraction of association rules, such as the Kmeans and Apriori algorithms.

Our system uses an RDF ontology for representing the argument map of any dialogue, stored as RDF-triples in the Virtuoso repository. The user interface is designed with Web technologies, whereas the Server Tier is implemented with Servlets and Java API classes.
Τα κοινωνικά δίκτυα εξελίσσονται συνεχώς για να υποστηρίξουν τις αυξανόμενες ανάγκες της κοινωνίας για ανταλλαγή γνώσεων, την αλληλεπίδραση και τη συνεργασία μεταξύ ανθρώπων. Αυτά τα δίκτυα παρέχουν διάφορες πλατφόρμες αλληλεπίδρασης μέσω των οποίων μπορούν να μοιράσουν τις απόψεις και τις εμπειρίες τους. Τα συστήματα διαλόγων είναι μερικά από αυτά τα συστήματα που επιτρέπουν στους χρήστες να εκφράσουν τις απόψεις τους με την μορφή επιχειρημάτων και να συμμετέχουν υπέρ ή κατά ιδεών που προκύπτουν μέσα από τον διάλογο με έναν πιο δομημένο τρόπο. Όσο τα κοινωνικά δίκτυα και άλλες τυχερές διαλόγους παράγουν όλο και περισσότερο περιεχόμενο, τόσο μεγαλύτερη ανάγκη υπάρχει για κατανόηση και ομαδοποίηση των απόψεων που έχουν εκφραστεί στα πλαίσια ενός διαλόγου, έτσι ώστε οι χρήστες να μην χρειάζεται να κάνουν ανάγνωση ολόκληρων συζητήσεων.

Το σύστημα παρέχει ένα περιβάλλον διαλόγου που στόχο έχει να παρακινήσει τους ανθρώπους να συμμετέχουν σε πιο στοχευμένους δομημένους διαλόγους. Σαν πλατφόρμα διαλόγου επιτρέπει στους χρήστες να δημιουργήσουν καινούργιες ιδέες, να κάνουν τις δικές τους ερωτήσεις, να υποβάλλουν υπέρ ή κατά επιχειρήματα, να σχολιάσουν και να ψηφίσουν. Ο γενικός στόχος είναι να προσφέρουμε διάφορες τροποποιήσεις στην ανάλυση των συζητήσεων, προσεκυνούμε να συμμετέχουμε μια πλήρη εικόνα για την εγκυρότητα, την αξιολόγηση και την πληρότητα των επιχειρήματος που έχουν διατυπωθεί μέσα από συζητήσεις.

Στην παρούσα δουλειά, ξεκινάμε παρέχοντας ένα σχήμα αναπαράστασης των συζητήσεων και των σχέσεων μεταξύ τους με δομημένα στοιχεία διαλόγου όπως είναι, θέμα (issue), τοποθέτηση (position), υποστηρικτικό επιχείρημα (pro-argument) και αντιπαράθεση (con-argument). Στην αναλυτική, εφαρμόζουμε μία τυπική μέθοδο αξιολόγησης των επιχειρημάτων που έχουν διατυπωθεί μέσα από συζητήσεις. Στο σύστημα προσφέρουμε μία σειρά από την παρούσα δουλειά, από τις οποίες οι πιο σημαντικές αφορούν την δημιουργία καινούργιων θεμάτων προς συζήτηση και την ανάλυση διαφορών πυχών του διαλόγου.

Στην παρούσα δουλειά, ξεκινάμε παρέχοντας ένα σχήμα αναπαράστασης των συζητήσεων και των σχέσεων μεταξύ τους με δομημένα στοιχεία διαλόγου όπως είναι, θέμα (issue), τοποθέτηση (position), υποστηρικτικό επιχείρημα (pro-argument) και αντιπαράθεση (con-argument). Στην αναλυτική, εφαρμόζουμε μία πιθανή μέθοδο αξιολόγησης των επιχειρημάτων, που ονομάζεται sim-Dice. Το βάρος του κάθε επιχειρήματος υπολογίζεται με βάση κάποιες μετρικές αξιολόγησης. Κύριος τρόπος επικεντρωνόμαστε στην προσέγγιση της ανάλυσης συζητήσεων λαμβάνοντας υπόψη διάφορες πυχών ενός διαλόγου. Η προσέγγιση αυτή καλύπτει σημαντικές πληροφορικές ανάγκες που προκύπτουν από τους χρήστες, προκειμένου να υποβληθούν συζητήσεις στα επιχειρήματα ανάλογα με την συσχέτιση των διάφορων χαρακτηριστικών του διαλόγου και την εξόρυξη κανόνων συσχέτισης όπως είναι οι Kmeans και Apriori αλγόριθμοι.

Το σύστημα χρησιμοποιεί για την αναπαράσταση διαλόγων μία RDF οντολογία, η οποία αποθηκεύεται στο Virtuoso με τη μορφή RDF τριπλετών. Η διεπαφή του
συστήματος έχει σχεδιαστεί με τεχνολογίες Web ενώ ο διακομιστής έχει αναπτύξει χρησιμοποιώντας Servlets και Java API classes.
Acknowledgments

First of all, i would like to thank my advisor, Professor Dimitris Plexousakis, for his support and trust on my abilities that gave me the confidence to continue when things were not good enough for me. I am also grateful to him for his expert guidance and his substantial contribution to the completion of this master thesis.

I would also like to express many thanks to, Assistant Professor Yannis Tzitzikas and Researcher Giorgos Flouris for their willingness to participate as members of committee in my master thesis presentation.

A special thanks should also go to Giorgos Flouris, Thodoris Patkos and Giannis Roussakis because i feel fortunate to have collaborated and discussed relevant matters with them. They were always there whenever their help was needed. Their precious advice and patience helped me to overcome every difficult situation and mostly to surpass my insecurity. I also appreciate the fact that i could rely on them protection from any eventuality that would delay or distract me from my goals.

Furthemore, I would like to acknowledge the support of the Institute of Computer Science (ICS-FORTH) both financially and in facilities, as well as the University of Crete throughout my graduate studies. Many thanks go to several people in ICS and the University for helping me in every concern and making me feel part of this academic community. Also, i would like to thank some of my special friends (i believe they know who they are) for being there to remind me that whatever happens in life, they will be there to encourage me in every step of my life.

This master thesis, would not have been possible without the unconditional love, support and encouragement from my family and specifically from my father Zenel, my mother Burbuqe, and my brother Aris , who is the most fervent supporter in my every effort. They were always there to inspire me through the rainy days. Their help was both invaluable and irreplaceable.

Last but not least, i would like to thank all my relatives for advising and encouraging me whenever was needed.

Many thanks to all of you.
This master thesis is dedicated to my parents, Zenel and Burbuqe
## Contents

Table of Contents i

List of Tables iii

List of Figures v

List of Algorithms vii

1 Introduction 1
   1.1 Contributions ........................................... 2
   1.2 Illustrative example .................................... 3
   1.3 Thesis Overview .......................................... 4

2 Background 7
   2.1 Argumentation theory .................................... 7
      2.1.1 Types of Argumentation .......................... 8
      2.1.2 Argument Representation .......................... 8
   2.2 The Argument Web ........................................ 10
      2.2.1 Argument Web infrastructure and architecture .... 11
   2.3 The Social Web ........................................... 11
   2.4 Issue-Based Information System (IBIS) ................. 12
   2.5 Clustering algorithms ................................... 13
      2.5.1 Kmeans Algorithm ................................. 13
      2.5.2 Description: Kmeans Algorithm Steps ............ 14
      2.5.3 Distance Similarity Measure ..................... 15
      2.5.4 Expectation-Maximization (EM) algorithm .......... 15
   2.6 Association Rule Learning Algorithms .................. 16
      2.6.1 Apriori .............................................. 16
         2.6.1.1 Two-Step Approach ............................ 16
      2.7 RDF and RDFS schema .................................. 17
      2.8 SPARQL ................................................. 17
3 Related Work 19
  3.1 Introduction ................................................. 19
      3.1.1 Argumentation tools ................................. 19
      3.1.2 Debating platforms ................................. 23
      3.1.3 Question and Answer websites (Q & A) ............... 24
      3.1.4 Visualization tools ................................. 25
      3.1.5 Synopsis ............................................. 26
      3.1.6 A taxonomy of the existing platforms ................. 27

4 Methodology 31
  4.1 Introduction ................................................. 31
  4.2 Basic Concepts ............................................. 32
      4.2.1 Concepts Definition of our mapping schema ............. 32
      4.2.2 WEKA: A collection of ML Algorithms used in APOPSIS .... 33
  4.3 Mapping Schema ............................................. 33
      4.3.1 Theoretical model .................................... 33
          4.3.1.1 Comparison between IBIS and our mapping schema .. 34
      4.3.2 Semantic Web Ontology domain ........................ 34
          4.3.2.1 Describing the conceptual components of Semantic Web Ontology ............................................. 35
  4.4 Our approach ................................................. 38
      4.4.1 s-mDiCE: A formal framework for evaluating arguments .... 39
      4.4.2 Opinions Analysis .................................... 40
          4.4.2.1 Describing algorithms ............................. 40
          4.4.2.2 Describing the "Best Practices" .................. 43

5 Implementation 49
  5.1 Apopsis: A debating platform ................................ 49
  5.2 Functionality .............................................. 50
      5.2.1 Logging mechanism .................................... 51
      5.2.2 Creating new debates .................................. 51
      5.2.3 Creating new positions ................................ 52
      5.2.4 Creating new arguments: supporting/attacking arguments .... 52
      5.2.5 Vote up/down on positions/arguments ................... 53
      5.2.6 Searching mechanism: retrieving supporting/attacking arguments ............................................. 53
      5.2.7 Two phases of dialogues ................................ 55
  5.3 Technical Information and Architecture ...................... 55
      5.3.1 Openlink Virtuoso Universal Server .................... 56

6 A System Use Case 57
  6.1 Main features of APOPSIS platform ........................ 57
      6.1.1 Two-stages of dialogue ............................... 57
      6.1.2 Creating new debates .................................. 58
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Six Basic Types of Dialog</td>
<td>9</td>
</tr>
<tr>
<td>3.1</td>
<td>Comparing existing tools(1)</td>
<td>28</td>
</tr>
<tr>
<td>3.2</td>
<td>Comparing existing tools(2)</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>K-means parameters</td>
<td>41</td>
</tr>
<tr>
<td>4.2</td>
<td>EM parameters</td>
<td>42</td>
</tr>
<tr>
<td>4.3</td>
<td>Apriori parameters</td>
<td>42</td>
</tr>
<tr>
<td>4.4</td>
<td>Users with similar preferences</td>
<td>43</td>
</tr>
<tr>
<td>4.5</td>
<td>Information need (1): Input/Output</td>
<td>44</td>
</tr>
<tr>
<td>4.6</td>
<td>Sharing similar views with others</td>
<td>44</td>
</tr>
<tr>
<td>4.7</td>
<td>Information need (2): Input/Output</td>
<td>44</td>
</tr>
<tr>
<td>4.8</td>
<td>Same users with similar views</td>
<td>45</td>
</tr>
<tr>
<td>4.9</td>
<td>Information need (3): Input/Output</td>
<td>45</td>
</tr>
<tr>
<td>4.10</td>
<td>Users with different views</td>
<td>45</td>
</tr>
<tr>
<td>4.11</td>
<td>Information need (4): Input/Output</td>
<td>46</td>
</tr>
<tr>
<td>4.12</td>
<td>Positions according to similar preferences</td>
<td>46</td>
</tr>
<tr>
<td>4.13</td>
<td>Information need (5): Input/Output</td>
<td>47</td>
</tr>
<tr>
<td>4.14</td>
<td>Topics based on users views</td>
<td>47</td>
</tr>
<tr>
<td>4.15</td>
<td>Information need (6): Input/Output</td>
<td>48</td>
</tr>
<tr>
<td>5.1</td>
<td>User Profile Characteristics</td>
<td>51</td>
</tr>
<tr>
<td>5.2</td>
<td>Searching type 1</td>
<td>54</td>
</tr>
<tr>
<td>5.3</td>
<td>Searching type 2</td>
<td>54</td>
</tr>
<tr>
<td>5.4</td>
<td>Searching type 3</td>
<td>54</td>
</tr>
<tr>
<td>5.5</td>
<td>Searching type 4</td>
<td>54</td>
</tr>
<tr>
<td>5.6</td>
<td>Searching type 5</td>
<td>54</td>
</tr>
<tr>
<td>5.7</td>
<td>Searching type 6</td>
<td>55</td>
</tr>
<tr>
<td>5.8</td>
<td>Argumentation Tools (1)</td>
<td>83</td>
</tr>
<tr>
<td>5.9</td>
<td>Argumentation Tools (2)</td>
<td>84</td>
</tr>
<tr>
<td>5.10</td>
<td>Argumentation Tools (3)</td>
<td>85</td>
</tr>
<tr>
<td>5.11</td>
<td>Argumentation Tools (4)</td>
<td>86</td>
</tr>
<tr>
<td>5.12</td>
<td>Argumentation Tools (5)</td>
<td>87</td>
</tr>
<tr>
<td>5.13</td>
<td>Argumentation Tools (6)</td>
<td>88</td>
</tr>
</tbody>
</table>
# List of Figures

2.1 An example that illustrates the Argument Web. ........................................... 10  
2.2 IBIS model. ................................................................................................. 13  
2.3 Sum of the Squared Error (SSE). ............................................................... 15  
2.4 Euclidean Distance Measure. ....................................................................... 15  
4.1 Argument Map. ............................................................................................. 34  
4.2 Semantic Web Ontology. ............................................................................... 35  
4.3 (a) A debate graph with votes, base score and user-generated supporting and attacking arguments, (b) its transformation with black nodes. ......................................................... 39  
5.1 The APOPSIS system. .................................................................................... 50  
5.2 Registration form. .......................................................................................... 51  
5.3 Login form. .................................................................................................... 51  
5.4 Creating new debate form. ............................................................................ 52  
5.5 Creating new position form. ......................................................................... 52  
5.6 A position argument......................................................................................... 52  
5.7 Creating new argument. ................................................................................. 53  
5.8 Supporting/attacking arguments. ................................................................. 53  
5.9 Vote on positions/arguments. ....................................................................... 53  
5.10 Types of searching mechanism. .................................................................... 54  
5.11 Basic architecture of Apopsis. ...................................................................... 56  
6.1 An instance of creating a new debate. ......................................................... 57  
6.2 An instance of creating a new debate. ............................................................ 58  
6.3 An instance of creating a new position. ....................................................... 59  
6.4 An instance of creating a new argument. ..................................................... 59  
6.5 An instance of voting negative. ..................................................................... 59  
6.6 An instance of evaluating an attack argument. ............................................. 59  
6.7 Query : Users with similar preferences. ....................................................... 60  
6.8 Results : Users with similar preferences. ...................................................... 60  
6.9 Clustering Query : Sharing similar views with others. ............................... 61  
6.10 Results : Sharing similar views with others. .............................................. 62  
6.11 Clustering Query : Same users with similar views. ................................... 62
6.12 Results: Some users with similar views. ................. 63
6.13 Clustering Query: Users with different views. .......... 63
6.14 Results: Users with different views. .................. 64
6.15 Clustering Query: Positions according to similar preferences. . 65
6.16 Results: Positions according to similar preferences. .... 65
6.17 Clustering Query: Topics based on users views. ........ 66
6.18 Results: Topics based on users views. .................. 66
6.19 An instance of searching type 1. ....................... 67
6.20 Results of searching type 1. ............................ 68
6.21 An instance of searching type 2. ....................... 68
6.22 Results of searching type 2. ............................ 69
6.23 An instance of searching type 3. ....................... 69
6.24 Results of searching type 3. ............................ 70
6.25 An instance of searching type 4. ....................... 70
6.26 Results of searching type 4. ............................ 71
6.27 An instance of searching type 5. ....................... 71
6.28 Results of searching type 5. ............................ 72
6.29 An instance of searching type 6. ....................... 72
6.30 Results of searching type 6. ............................ 73

1   Classes Hierarchy. .................................. 90
2   Creation Data. ...................................... 91
3   Strength. ........................................... 91
4   Content. ........................................... 91
5   Issue. ............................................. 92
6   Topic. ............................................. 92
7   Position. .......................................... 92
8   Argument. .......................................... 93
9   Comment. .......................................... 93
10  Rating. ............................................. 93
11  Vote. ............................................... 94
12  Star Based. ........................................ 94
13  Category. .......................................... 94
14  Proposition. ........................................ 95
15  Properties Hierarchy. ............................... 95
16  has content value property. .......................... 96
17  has content type property. .......................... 96
18  datetime property. .................................. 97
19  voting property. .................................... 97
20  dialogue property. ................................... 97
21  congruence property. ................................ 98
22  acceptance property. ................................ 98
23  quality property. ................................... 98
24  created by property. ................................ 99
List of Algorithms

1. Kmeans Pseudo-Code ........................................ 14
2. Frequent itemset generation of the Apriori Algorithm ........ 17
Chapter 1

Introduction

New opinions are constantly presented every day on the World Wide Web, where users feel free to express their views and share their experiences on popular blogs, discussion forums and debate portals as well. One of the main achievements of the Web is that it has enabled users to argue on a variety of topics. The dialogues often hide very interesting facts and opinions. However, most dialogues encountered on the Web, are unstructured, meaningless and often chaotic, especially the dialogues on topical issues and topics of general interest that are of great interest, as there reflected public opinion. Because of the dynamic nature of debates and the large number of comments they contain, such dialogues, are difficult and time consuming to someone monitors them. Hence, it is essential and helpful to analyze and provide in a more comprehensible, readable and concise form in order to facilitate monitoring and their participation in these dialogues.

From the research point, Semantic Web [2] has led us in that direction as constitutes an appropriate means of representing the arguments uniformly on the Web and provide common languages and principles for modeling opinions in a structured way. This Web has an explicit semantic structure that connects the statements and arguments in a debate, allowing for far better navigation and analysis of a debate. Formalisms such as RDF [28] allow researchers facilitating conversations by proposing ways to adorn any content on the Web with metadata. This allows connecting contents on the Web, both via hyperlinks and semantic relations, so as to offer functionalities like creating new debates, commenting on existing arguments and evaluating the strength of each arguments within discussions. Hence, Semantic Web help in supporting argumentation in the form of identifying, resolving, representing and storing the arguments. In order the semantic features to be applied in conversations, a suitable ontology required that will represent the conceptual components of dialogue.

Furthermore, the need of describing the relations between opinions in discussions, such as "support" or "attack", results on implementing several algorithms that evaluate the strength of each argument by using different metrics, such as in
Apopsis\textsuperscript{1}, where a formal framework, called s-mDice is applied for the evaluation of arguments, considering both well-justified arguments and votes. Other frameworks that evaluate supporting and attacking arguments is the Extended Social Abstract Argumentation Framework (ESAAF) \cite{16}, including voting mechanism and arguments as well, applied in the Quaestio-it.com \cite{17} system.

Several software systems are developed for facilitating structured dialogues. These systems provide a collection of differing views where users can support their stance by stating arguments in a positive or negative way. Such platforms are the CreateDebate\textsuperscript{2} which is an opinions community, containing perspective debates, the Debate.org\textsuperscript{3}, a social network for creating and participating in debates. Another similar system is the iDebate\textsuperscript{4} which allows users to participate in a specific topic by providing statements for or against an initial position and eDialogos \cite{1}, a platform for motivating users to participate and vote about city planning policies. However, these systems do not apply any reliable evaluation method for the strength of arguments. Most of the systems focus on providing comments in support of or against a statement, provide questions and answers or goals to a debate outcome for a decision-making problems such as . Despite the fact that, they provide structured conversations, none of these systems is able to analyze opinions in order to assist decision makers in understanding the dynamics of communities.

Our system, Apopsis is a web-based system, implemented for analyzing and providing structured opinions. The purpose of Apopsis is not to come up with a solution to the topic discussion. Indeed, the system provides and displays the different opinions prevailing in debates throughout the decision-making process and extract the most relevant arguments expressed within discussions. Participants are helped by the system in understanding the various aspects of dialogues and in facilitating the effort of searching the most expressive opinions that share similar or different views with them.

\section{Contributions}

To the best of our knowledge, this work is the first study that focuses on the analysis of opinions, considering the different aspects of dialogues and users views expressed in the form of agreement and disagreement. Assuming a dialogue where more and more opinions are presented, the more difficult is to monitor the flow of the dialogue. Intuitive methodologies are needed for analyzing the different opinions presented every day on the debate platforms. Our goal is moving in that direction in order to provide opinions analysis that will help the audience to get a clear picture of the different views prevailing on a goal-oriented topic. Next, we describe the most important features that Apopsis supports.

\textsuperscript{1}http://www.ics.forth.gr/isl/Apopsis
\textsuperscript{2}http://www.createdebate.com
\textsuperscript{3}http://debate.org
\textsuperscript{4}http://www.idebate.org
• A combination of machine learning algorithms is used for the analysis of users opinions, including clustering algorithms for the clustering of features and associations rules for the extraction of interesting rules.

• The system applies an existing formal framework, called sm-Dice, for evaluating the strength value of each argument which is based on a multi-aspect evaluation.

• An RDF ontology is used for representing the structure of knowledge domains that stores and retrieves discussions in form of RDF statements.

Moreover, the system provides a range of functionalities with the purpose of encouraging users to participate and collaborate in dialogues, expressing their agreement and disagreement. These features are shown below.

• Logging mechanism.

• Creating new topics of discussion.

• Creating new position/arguments.

• Retrieving supporting and attacking arguments.

• Two different phases: Debates proceed in two different stages, which are facilitated by a class of users, known as "moderators", ensuring the high and quality analysis of opinions.

1.2 Illustrative example

At this point, we are going to give an example, enhancing the analysis of opinions within dialogues and assisting the decision making.

Consider a city, where the City Council needs to take some important decisions about the planning process and the implementation of the city’s actions, enabling residents to be involved in dialogues through debate platforms where they can participate by expressing their agreement or disagreement and vote on goal-oriented topics of the city. In our example, the debate platform aims at giving the opportunity to the citizens of the municipality, to work together for designing and implementing policies for the municipality. A system, such as Apopsis, could assist decision makers in understanding the dynamics of local communities, by identifying the common characteristics among people sharing similar opinions. As decisions made by city councils are rarely based on a black-or-white basis, it is important to identify the different trends and driving forces of the various groups that are formed, in order to try to accommodate as many of their needs as possible.

The consultation process is based on the principle that people involved in dialogues, help the City Council make decisions on specific problems and issues.
CHAPTER 1. INTRODUCTION

according to how the citizens opinions are distributed and analyzed throughout the decision making process. Furthermore, the residents and other groups will be informed about the methods and regulations of the city.

The dialogue starts with the City Council suggesting new topics for consultation and involving a variety of groups such as citizens, groups of civil society, the municipality and other bodies with a view of deciding on the effective governance of the city. Let’s assume that, the topic of discussion is the regeneration of the city center. Then, every resident participates by providing solutions or ideas, known as positions, which respond to the open issue. Depending on the users beliefs, agreement and disagreement are presented within conversations. For instance, a part of conversation may include opinions that agree with the regeneration of the city with the purpose of improving significant problems that city faces. On the other hand, there are opinions that disagree with the city Council’s suggestion as the financial situation of country makes it impossible to regenerate the city. The dialogue continues until a significant number of participants and well-justified comments have arisen. Then, the best arguments, according to the evaluation algorithm and existing moderators who ensure for the quality of opinions, are considered for debate analysis.

At the end, the system help both groups of participants and city council to be informed for the most important opinions of city council’s suggestions where they can identify the similar or different views prevailing in the dialogue.

1.3 Thesis Overview

The rest of this work is organized in six main chapters.

• Chapter 2, provides the relevant background and describes the main prerequisites.

• Chapter 3, provides a review of the related work, describing all argumentation tools and debate platforms that are closely related to our debate platform, Apopsis. Furthermore, we make a comparison among existing software tools and our debate portal

• Chapter 4 constitutes the main core of this master thesis: (i) introduces the basic concept of our methodology, (ii) describes the ontology and its conceptual components and finally (iii) elaborates on the methodology we used for analyzing the various aspects of users opinions.

• Chapter 5 analyzes in detail the functionality of our system and its architecture.

• In Chapter 6 we provide 2 scenarios, where the platform would be applicable in order to discuss various design decisions and issues, arising within Apopsis system
1.3. THESIS OVERVIEW

• The last chapter draws some conclusions about this work and identifies issues for further research.

Our system, Apopsis is available in the following URL, http://www.ics.forth.gr/isl/Apopsis
CHAPTER 1. INTRODUCTION
Chapter 2

Background

2.1 Argumentation theory

Argumentation is defined as “a verbal and social activity of reason aimed at increasing (or decreasing) the acceptability of a controversial standpoint for the listener or reader, by putting forward a constellation of propositions intended to justify (or refute) the standpoint before a rational judge”. It is the field of study in which rhetoric, logic and dialectic meet (Rahwan et al. 2007b) [48]. Argumentation constitutes a major component of humans intelligence and involves the identification, analysis and evaluation of arguments. The goal is to persuade or convince that one reasoning is more valid or appropriate. As a rich research area is applicable in various field like artificial intelligence, collaborative learning, philosophy, linguistics and psychology. Represent the study of agreement and disagreement that humans express with the goal of defending their opinions or convincing their selves and others in everyday life. The ability to engage in arguments is essential for people to understand new problems, perform practical reasoning and justifying beliefs. Arguments on the Web can be used in decision-making contexts. People may agree or disagree on the facts of situation on the preferred outcome or on the decision that need to be taken. Not all people have the ability to be good arguers. For that reason, several argumentation tools have been implemented by researchers, evolving people in several types of discussions which can help them to facilitate communication and argumentation among multiple participants. These argumentation systems are mostly used for visualizing, evaluating and searching arguments in structured dialogues. We mostly focus on argumentative discussions, which consider online textual messages and discussions where users opinions are important and relevant. Such argumentative discussions can be found in many online platforms such as open source communities where people are collaborating in support of decision-making process. Much research effort has been spent in defining primitive structure for arguments as well as for the types of interaction between them. Ontologies are used for describing and making users ware of the
conceptual components in the task of specific domain. In this point, common languages and principles are provided to model and query information on the Web such as RDF [28], RDFS [41] and SPARQL [45].

2.1.1 Types of Argumentation

Several types of argumentation exist depending on the context and goals of each dialogue. Walton in [69] made a classification of the dialogue into six categories: Persuasion, Inquiry, Negotiation, Information Seeking, Deliberation and Eristic. Persuasion aims at resolving a difference between opinions, Inquiry is aimed at "growth of knowledge and agreement", Negotiation tries to resolve a conflict of interest by reaching a deal, information seeking aims at exchanging information among participants into dialogues, deliberation tries to reach a decision on a course of action and finally Eristic is the verbal constitute of a fight. We are mostly interested in persuasion type of argumentation, where the Apopsis system supports opinions that are presented as well-documented viewpoints about topic that aims to convince some audience.

**Persuasion**: Persuasion is the process of exchanging arguments and opinions among people who are engaged into deliberation dialogues. The ability of an argument to convince someone depends on the data evidence that are used by agents. This type of argumentation focuses on conflict resolution problems mostly in the domain of multi-agent systems. The goal for an agent who engaged in a persuasive dialogue is to convince other contributors about his opinion using either his mental beliefs either by exploiting the information that is coming from the knowledge and beliefs of others in order to use it for persuading other people. When there are not enough data to convince someone, agents use any information about users profile characteristics, so their arguments will cause more impact into dialogues. An interesting problem here is to identify the set of characteristics that will be useful to use in persuasion. These set of characteristics mostly include agents beliefs, their interests and general preferences. An important feature that should be taken into consider is the context of argumentation as different data and evidence can be used for convincing someone. To make an assessment of how convincing an argument is, functions and models that evaluate the strength of each argument need to be suggested. But in most cases, the strength of each argument may differ for different audiences or context because of the impact of argument.

2.1.2 Argument Representation

Argument is the elementary unit of the argumentation process so its construction must be simple enough and the same time as general as possible, in order to support many different domains and types of argumentation. Arguments are presented every day on the web, in discussion forums, blogs and many other social communities. As result, the Web act as an enabler for a large-scale argumentation
## 2.1. ARGUMENTATION THEORY

<table>
<thead>
<tr>
<th>Type of Dialog</th>
<th>Initial situation</th>
<th>Participant’s goal</th>
<th>Goal of dialog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Persuasion</td>
<td>Conflict of Opinions</td>
<td>Persuade Other Party</td>
<td>Resolve or Clarify Issue</td>
</tr>
<tr>
<td>Inquiry</td>
<td>Need to have Proof</td>
<td>Find and Verify Evidence</td>
<td>Prove (Disprove) Hypothesis</td>
</tr>
<tr>
<td>Negotation</td>
<td>Conflict of Interests</td>
<td>Get What You Must Want</td>
<td>Reasonable Settlement Both Can Live With</td>
</tr>
<tr>
<td>Information-Seeking</td>
<td>Need Information</td>
<td>Acquire or Give Information</td>
<td>Exchange Information</td>
</tr>
<tr>
<td>Deliberation</td>
<td>Dilemma or Practical Choice</td>
<td>Coordinate Goals and Actions</td>
<td>Decide Best Available Course of Action</td>
</tr>
<tr>
<td>Eristic</td>
<td>Personal Conflict</td>
<td>Verbally Hit Out at Opponent</td>
<td>Reveal Deeper Bias of Conflict</td>
</tr>
</tbody>
</table>

Table 2.1: Six Basic Types of Dialog

where different opinions are presented and evaluated by contributors and readers. The main goal of argumentation technology is to provide an argument representation that allow users to create, manipulate and review arguments. Many software tools have been implemented for analyzing textual arguments, for constructing well-organized textual arguments and for organizing arguments into debates on the Web.

Several representation formats have been used in existing systems. Some formats aim at supporting communication among users while others represent the structure of debate. We will mention the major representation types that found in the literature Suthers’ 2003 [65]: (1) Linear, (2) Threaded, (3) Graph-based, (4) Container and (5) Matrix.

1. **Linear.** The simplest form of argument is a linear, usually textual form. Opinions and arguments are expressed in the form of sequential in communication. The problem with these types of arguments is that when you are participate in dialogs, it is not clear which responses refer to which other comments.

2. **Threaded.** These discussions explicitly capture comments-reply sequences and support users participating in large argumentation strands. Many efforts have been done for annotating the role of contributions in an argument. In our case, we use a forum-like style of discussions, where contributions are marked as, for instance, issue, position, pro-argument and con-argument.

3. **Graph-based.** Graph-based representation style allow contributions to be displayed as nodes that represent the argument component. The edges represent the relations between the arguments.

4. **Container.** The goal of containers is to visualize argumentation strands belonging together graphically via frames.

5. **Matrix.** An attempt to visualize implicit or missing relations between argument elements where argument components are the rows and columns of the matrix while cells represent the relations between the components.
2.2 The Argument Web

The Argument Web is a large-scale Web of structured and inter-connected arguments. Represents the first technology linking debate, disagreement, and argument structures from a variety of tools and applications for creating, manipulating, extending, navigating and visualizing the Argument Web resources that make up the interconnected network applied in different domains. The Argument Web’s objective is to improve the quality of online arguments and debate that can make them intuitive for its contributors, including mediators, students, academic and bloggers. By increasing the Web resources the community gain access to a valuable recourse. As a Web, provides infrastructures that allow the interconnection of argumentative viewpoints that can be posted anywhere through a underlying ontology of arguments. Such an ontology is based on the reification of the AIF which is a strong candidate for forming the foundation of Argument Web. Its specification can be extended to capture different argumentation formalism and schemes. This ontology assumes that arguments can be represented as nodes in directed graph called as argument network. A number of tools have been implemented as a part of the Argument Web that can offer support for explicit arguments and to improve large-scale debates on the Web. Several blogs, unstructured or structured online discussion forums can provide the communication in a fairly satisfactory manner. However, many of these tools are all self-contained, and the arguments represented in them cannot be interchanged. For that reason, next it is shown an example that illustrates the Argument Web and showing how the arguments are interconnected.

The problem we face in this example is Assad’s Syria issue and the morality of potential Western intervention. You query the Web by asking the question: Should we invade Syria? You are presented with various sources such as arguments that

![Figure 2.1: An example that illustrates the Argument Web.](image-url)
2.3. THE SOCIAL WEB

links to other sources, videos, newspapers columns and many articles, discussing the Syria issue. Focusing on the newspaper column, you can get another page that links to other arguments posted on the web that agree or disagree with the fact that Syria is different from Iraq. You inspect the counterarguments in this article and expect that there exist another page that show you the total disagreement of this issue. In this way, the Argument Web makes it possible to follow a line of argument across the user comments, multimedia resources and so on. Links tagged pro indicate that one statement agrees with another statement, which means that one statement provides a reason for the other statement. On the other hand, con links indicate that a counterargument is provided, which means that one statement is a reason against another statement.

2.2.1 Argument Web infrastructure and architecture

Argument Web is a large-scale deployment of Semantic Web Technology that aims to create a Web infrastructure that allows for storage and automatic retrieval and analysis of linked argument data. It is based on a common ontology for argument called the Argument Interchange Format (AIF) [9]. Moreover, the Web approach builds on highly scalable database systems, which is still conforming to the main principle of Linked Data, each argument is addressable by a unique URI. So an entity in the Argument Web can be accessed both by Semantic Web technology as it is expressed in RDF and by a range of web services written in PHP which allow the addition and retrieval of AIF components from the database to import and export argument data in a range of widely used formats. Furthermore, a middle layer exists which groups web service queries that combine information for both structure and rules of argumentative dialogue.

2.3 The Social Web

A Social Web is the way in which people are developing their communication with others and forming relationships of different types among them. More specifically, the Social Web known also as Social Networks have the ability to brings people close to each other, share their ideas and make friends through Web interaction. By using the term of Social Networks we mean a web-based services that allow individuals to construct a public profile, to view and express their viewpoints. There also exist the notion of debating in variety of topics and different domains where participants can express their opinions in support of or against others. Humans argue for a variety of reason and in most cases the problem still remain unsolved. This kind of arguing is quite important in the Social Web as can connect people via social software tools which means that plethora of social applications are necessary to be developed that can provide structured dialogues on the Web. What makes social networks unique is not that they are knowing new people and are connected with each other publicly but they allow individuals
to participate into structured debates where they can support other arguments or vote up/down on arguments. Among these people exist experts or just enthusiasts that want to debate in particular debate topics in order to influence the outcome of discussion about "wicked problems" that need to be solved or simply follow the debate discussion. The theory of argumentation schemes play an important role when we want to formalize and structure the online discussions, as well as users opinions. For understanding better the Social Web and how people can be part of it, we provide an example next showing a social network of sales and reviews.

Suppose we have a social network, such as eBay or Amazon, that run commercial transactions between its members and inform about them. Therefore, buyers and sellers can bye or sell the products and leave feedback about these commercial transactions. The type of dialogue enabled in this case is information seeking and sharing by leaving feedback and persuasion by supporting this feedback with well-justified comments. Suppose now that our social network consists of three users only, the first user (User1) is the seller and the others two are buyers (User2 and User3). The first user(User1) sold a product to User2 and User3, while the User2 provides a feedback to User1 about his sale. Users can also leave comments about the transaction that was done in the form of "good comments" or "bad comments". Furthermore, they can vote the answers and the social network can aggregates the reviews which represent a type of measure for the reputation that the product has received. In this point, a structured discussion among sellers and buyers have been developed that allow them to express their opinions in the form of agreements or disagreements about the products and voting them as well.

2.4 Issue-Based Information System (IBIS)

IBIS, Issue-Based Information System, is a problem-solving structure that was published in 1970 [29]. As a theoretical model is focused around controversial issues which take the form of questions. IBIS is designed to organize discussions and allow subsequent understanding of the decision taken and mostly is used to support communities and political decision-making. The ontology of this framework consists of questions, positions, pro-argument and con-argument. Issue is the most important part of this model, as it allows a thread discussion to be start. Furthermore, the positions are the solutions or ideas to the question that is given and the arguments that can be either positive or negative, represent opinions, facts and data. Several types of relations exist between issues such as, direct successor, generalization, relevant analogy, compatible, consistent, and inconsistent. This ontology, influences several others ontologies and systems such as debate systems. Next, we show the IBIS model.
2.5 Clustering algorithms

Clustering analysis has been an emerging research issue for a variety of applications. It is defined as the process of partitioning a set of data (or objects) into a set of meaningful sub-classes, called clusters. The goal of clustering is to group similar objects into classes on a given features and constitutes the most important unsupervised learning problem. A "good" clustering will produce highly clusters which means that the similarity of object into classes is high and the similarity of different classes is low. For that reason, the quality of clusters depends on the similarity measure that is used in clustering. The ability of discovering some or all hidden pattern is another measure that is related with the quality of clusters. Clustering can be achieved by various algorithms. One such algorithm is the Kmeans algorithm which we have used in this master thesis for analyzing the different aspects of dialogue. The algorithm is described in the following subsection.

2.5.1 Kmeans Algorithm

Kmeans is one of the simplest and most commonly used unsupervised learning algorithm that solves the clustering problem. The goal of Kmeans algorithm is to minimize the sum of the squared distances to the clusters centers. The procedure follows a simple way of classifying a set of given data through a set of number of clusters. This algorithm is adapted to many problem domains and can be applied to many field such as Marketing, Insurance, Earthquake studies, WWW, etc. Kmeans algorithm is composed of the following steps using pseudo-code.
Algorithm 1 Kmeans Pseudo-Code

1: procedure K-MEANS
2:   Select K points as the initial centroids.
3:   repeat
4:     Form K clusters by assigning all points to the closest centroid.
5:     Recompute the centroid of each cluster.
6:   until The centroids do not change.

2.5.2 Description: Kmeans Algorithm Steps

K-means algorithm is randomly selecting K initial centroids where K is a user defined number of clusters in most cases but there are also clustering techniques that allow you to generate the appropriate clusters according to the distribution of data. Each point is assigned to the closest centroid and the collection of point close to this centroid. The centroids are updated according to the points that exist into clusters and the process continues until the clusters are not changing anymore.

Initialization Step: Initialization Step is the step of choosing the initial centroids. Multiple runs of algorithm can produce different results which in most cases are not satisfactorily. It is necessary to mention here that the choice of initial centroids is significantly affecting the results that are produced by running the K-means algorithm. There are different ways of selecting the initial centroids. The most commonly used are the following:

(1) Running the algorithm multiple times with different set of randomly chosen centroids and then select the number of clusters (K) with the minimum SSE (Sum of Squared Error).
(2) Take a sample of points and cluster them according to some clustering techniques that will be able to produce the appropriate number of centroids. Next, use these centroids as initial centroids to the K-means algorithm to produce the clustering.

Assignment Step: After choosing the initial centroids, we need to determine which points belong to the closest cluster according to particular data and the distance similarity measure.

Re-estimation Step: An important aspect of running K-means algorithm is that of re-estimating the initial centroids based on the K-means algorithm and the distance measure. There is a mathematical way of calculating the objective function for deciding in which cluster should points be assigned. The basic idea is to minimize the Sum of the Squared Error (SSE) which calculates for each point the distance from that point (centroid) to the nearest cluster, squares it and finally adds up the sum of all points in the data. Procedure stops when there are not centroids that can be changed. Next, is shown the SSE function.

where:
2.5. CLUSTERING ALGORITHMS

2.5.3 Distance Similarity Measure

There are many similarity measures that can be used with the K-means algorithm such as Manhattan, Euclidean and cosine distance. Euclidean distance is commonly used in clustering procedure for K-means algorithm. As a similarity measure, represents the degree of correspondence among objects across all of the characteristics that exist in data. Next, we show the Euclidean distance measure.

\[
\text{d}(\mathbf{p}, \mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.
\]

Figure 2.4: Euclidian Distance Measure.

2.5.4 Expectation-Maximization (EM) algorithm

The EM algorithm is an iterative probabilistic clustering algorithm. Each cluster is defined by probabilities for instances to have specific values for their attributes. The algorithm generates probabilistic descriptions of the clusters in terms of means and standard deviation for numeric values of attributes and generates probabilities values for nominal values of attributes. The EM algorithm is an efficient procedure to compute the Maximum Likelihood (ML) when we have missing or hidden data. EM can be used as a pre-processing procedure of the K-means algorithm and as algorithm can decide how many clusters to create by cross-validation. Each iteration of the EM algorithm consists of two processes, called as the E-step, and the M-step. The algorithm aims at finding a lower bound to the likelihood function, the E-Step, and then maximizing this bound in the M-Step.

- Expectation Step (E-step)
In the E-step procedure, the missing data are estimated given the observed data and current estimate of the model parameters. More specifically, this step estimate the distribution over data given a fixed model.
- Maximization Step (M-step)
In the M-step procedure, the likelihood function is maximized making the assumption that missing data are known. New parameters are chosen for model to maximize expected log-likelihood of observed data.

2.6 Association Rule Learning Algorithms

Association analysis is a methodology which is useful for discovering interesting relationships hidden in huge data. These relationships are presented in the form of associations rules. A rule suggests that strong relationships exist between the attributes. The strength of an association rule can be measured in terms of support and confidence. (1) Support determines how often a rule is applicable by a data set, while (2) confidence determines how frequently items in an Y item set appear in transactions that contains X, where X and Y are disjoint itemsets. These measures are very important for the results that we can obtain by running associations rule learning algorithms. Support has a huge affect when we come to interpret the results as rules with very low support may be occurred by chance. On the other hand, confidence measures the reliability of the inference made by a rule. Also, indicates as an estimation of the conditional probability of Y and X which were mentioned before. Finally, associations rules are applicable in different domains such as Web mining, medical diagnosis, business market data etc.

2.6.1 Apriori

Apriori is the first association rule mining algorithm that incorporated the use of support-based pruning to control the exponential growth of candidate itemsets. Apriori is designed to operate on database as transactions are, for instance, collections of items bought buy customers. Requires a priory knowledge to generate the next generation of itemsets i.e generates the itemset (k+1) from the k-itemset.

2.6.1.1 Two-Step Approach

- Frequent Itemset Generation
Generate all itemsets whose support is bigger than minsup.

- Rule Generation
Generate high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset.
Algorithm 2 Frequent itemset generation of the Apriori Algorithm

1: \( C_k \): Candidate itemset of size \( k \)
2: \( L_k \): frequent itemset of size \( k \)
3: \( L_1 \): frequent itemset
4: 
5: for each node \( k = 1; k \neq 0; k + + \) do
6: \( C_{k+1} \) = Candidates generated from \( L_k \).
7: for each transaction \( t \) in database do
8: \((k + +)\) all candidates in \( C_{k+1} \) that are contained in \( t \).
9: \( L_{k+1} \) = Candidates in \( C_{k+1} \) with min support
10: return \( U_k \) \( L_k \);

2.7 RDF and RDFs schema

The Resource Description Framework (RDF) [28] is a simple language for expressing data models, which refers to resources (web resources). As a meta-data model is based on the idea of making statements and resources. A resource has the form of triplet and has a unique Universal Resource Identifier (URI) that can be considered as an entity, for instance, a file or an image. A collection of RDF statements represents a labeled, directed multi-graph. The elements of a triple are called, subject, predicate and object. The subject indicates the resource that is described by the statement. The predicate describes the relationship between the subject and the object. Finally, the object can be either a URI or a literal (e.g text). There are many different formats that RDF statements can be represented. These variety of formats consist of RDF/XML, Notation3, Turtle, NTriples and RDFa. In this master thesis we mostly focus on RDF/XML format.

RDF Schema (RDFS) [41] is an XML based knowledge representation language which it is written in RDF and as a vocabulary describes properties and classes of RDF-based resources. RDFS constitutes a semantic-extension of RDF. It provides mechanism for describing related resources and their relationships. Furthermore, the statements of RDF Schema (RDFS) make it possible to define hierarchies of classes and properties as well as to describe domains and ranges of the properties.

2.8 SPARQL

SPARQL [45] is the standard RDF query language for semantic web data resources that can retrieve and manipulate data resources stored in RDF format. SPARQL can be used to express queries across data resources, when the data is stored as RDF or viewed as RDF through middleware. In general, most formats of SPARQL query contain a set of triple patterns called a graph pattern. As a
language, provide fine-grained data access. Also, determines the patterns to seek for, when the answer form RDF graph matches this pattern.
Chapter 3

Related Work

This section examines the state of the art tools with similar orientation and points out their differences in relation to our work.

3.1 Introduction

Several software platforms are developed to serve the need of querying, searching, analyzing, evaluating and visualizing arguments in an informative and interactive way. Argumentation tools have gained attention in e-government and education mostly. In this introduction, we provide a brief overview of some existing tools that are close to our work. Furthermore, we determine the purpose of every tool and provide a comparison of the most important features among several tools. For a comprehensive investigation of Argumentation and Social Web tools, we relied on the following works (Schneider, Jodi and Groza, Tudor and Passant, Alexandre, 2013) [57] and (Scheuer, Oliver and Loll, Frank and Pinkwart, Niels and McLaren, Bruce M, 2010) [55].

Finally, we distinguish 4 different types of implemented software tools such as, (i) Argumentation tools, (ii) Debating platforms, (iii) Question and Answer websites (Q & A) and (iv) Visualization tools.

3.1.1 Argumentation tools

Araucaria [52] is a repository of arguments drawn from newspaper editorials, parliamentary reports and judicial summaries. It is chiefly intended for pedagogical use addressing the need to improve students’ critical thinking. Using argumentation schemes, the result of any given analysis is a marked up version of the original text. The software supports several different diagramming methods, including Toulmin diagrams and the Beardsley/Freeman ”standard” diagramming method. The format of this markup is described by the Argument Markup Language, AML, described in detail in [51]. The Araucaria System that creates files marked up according to AML is a tool of informal logic. Furthermore, there is not the notion
of attack and conflicting arguments and as a result there is no mechanism that evaluates arguments according to their acceptability. As such, it can be employed as an aid to support argument analysis and as a diagrammatic presentation tool.

**Reason!Able** [67] is an educational software that supports argument mapping to teach reasoning skills. It provides support to the users by guiding them step-by-step through the construction process. The argument trees constructed by Reason! Able contain claims, reasons, and objections. The software offers the possibility to build and manipulate representations of arguments, and evaluate the arguments that have been built. The elements of arguments are statements - declarative sentences that can be true or false. Some statements are reasons for (or objections to) others. A well-formed argument has reasons that work together to support the conclusion, just as a proper objection consists of a set of statements that work together to undermine a conclusion. The hierarchical structure of the argument maps built in Reason! Able makes the relations between the components (claims that support or undermine this or that claim) visible and explicit through their spatial arrangement, and the lines linking the components.

**Athena**\(^1\) [53] is an argument mapping software that is designed to support analysis and production of reasoning and argumentation by students in higher education at undergraduate and postgraduate levels. The software was developed with the purpose of extracting methods and principles from argumentation and critical thinking. It was implemented using the independent language Java. The graphic tree structure of Athena uses nodes to represent the conclusions and the premises. Lines are used to represent the logical relations of an argument. The strength of arguments can be graphically evaluated using the concepts of “Acceptability” and “Relevance”. The educational level suitable for Athena is comparable to that of Reason! Able, built at the University of Melbourne. The software agrees in emphasizing that reasoning should be taught as a practice rather than the application of logical theory.

**Parmenides**\(^2\) (Atkinson et al., 2006) [3], [8], [7] is a web-based tool for deliberative democracy created with the aim of presenting political action to the public based on argument schemes and critical questions from argumentation theory. Allows effective participation of citizens in governmental decisions. Participants respond to a variety of questions by asking them whether they agree or disagree. At the end of the survey, users submit an alternative proposal, where are shown the answers they choose. The system analyzes the users positions, using argumentation techniques in favor of the government, giving a more perception of the public sentiment for the specific topic.

**TruthMapping**\(^3\). TruthMapping is a Web-based system for capturing conversations in a group. It helps users store their claims and support them with

---

\(^1\)http://www.athenasoft.org  
\(^2\)http://cgi.csc.liv.ac.uk/~parmenides  
\(^3\)http://www.truthmapping.com
3.1. **INTRODUCTION**

corresponding links. The tool supports the Beardsley approach to present the argument. It supports two types of the argument deductive and inductive argument. A state map visually summarizes the overall user rating of different parts of an argument. Basic keyword search is provided over categories and topics. There are different discussion categories such as education, media, politics, and science and so on. Participants can agree or disagree over a certain topic that has already been posted.

**Rationale** [68] is a commercial package allowing the diagramming and visualization of arguments. The system was not designed for legal reasoning. However, there is incipient use in law schools, legal training programs and ‘on the job’ in law firms. Rationale facilitates the creation of ‘box-and-arrow’ argument maps, where users link premise to conclusions with boxes and arrows.

**Deliberatorium** [26] is a conceptual argument mapping tool. Organize discussions based on the topic instead of the time. This tool used to debate about wicked problems such as climate change, sustainability and so on. The main objective of the tool is to enable better large-scale collaborative deliberation. The analysis in this tool is based on IBIS argument mapping model. The main four components of this tool are Issue, Idea, argument for (pro), argument against (con) and part of arg.

**ArgDF**[^70] [Rahwan et al., 2007] is a pilot system based on an RDFS ontology that models AIF specifications (Chesnevar et al., 2006) and extends it to include Walton’s account of argumentation schemes (Walton, 1996). ArgDB uses Sesame RDF for storage and querying and Phesame for communicating with the Sesame through PHP pages. In ArgDF users can author new arguments that adhere to any of the available schemes; they can attack or support existing arguments. Users can also extend the underlying ontology by adding new argumentation schemes. A semantic-based keyword search facility is also offered by the system that return the arguments (supporting/attacking) arguments of a claim containing a specific keyword.

**Avicenna** [47] is a Web-based system using Jena, ARQ and Pellet and is based on the OWL ontology ArgDF [70]. The basic functionalities it offers, are the chaining of arguments that is retrieving all arguments that directly or indirectly support a given conclusion and secondly that arguments can be classified into the hierarchy of argument schemes defined by Walton. It supports storage of arguments in persistent RDF storage where it is possible to create new arguments by introducing new claims, reusing existing claims or introducing new attacks. The system implements and uses queries in different tasks. Queries are also utilized to return attacking or supporting arguments of a given claim, searching for supporting arguments and retrieving arguments that support the claim both directly and indirectly.

[^70]: [http://www.argdf.org](http://www.argdf.org)
**DiscourseDB**\(^5\) is the online database of opinions and commentary. By making use of wiki technology, it collects opinions of journalists and commentators from various news sources. Users can search for opinions for or against a position, according to specific author or topic. As a system is useful for exploring and summarizing opinions but do not provide online debates among users. Argumentation theory and technology in terms of structuring the opinions or evaluating the positions are not taken into consideration.

**ConvinceMe**\(^6\) is an online competitive debating environment where users participate in real debates. Users are involved in one-on-one debates between two users, called as battles, where they add arguments in hopes of getting readers votes. The rest of users vote for the arguments that are more convincing to them and the winner of the debate is the one who gets the most votes. No structure is defined to the arguments and the counterarguments.

**Carneades**\(^7\) [19] is a set of Open Source software tools providing support for a range of argumentation tasks including evaluation, construction and visualization. It is particularly suitable for legal applications. Users through a front-end web application, enter facts of cases to send feedback to legislative bodies about policy issues and legislative proposals. The legal norms and policies are modeled with high-level declarative rule language. The same language is also used to model Walton’s argument schemes and critical questions, which are exploited for the classification of the arguments and for managing enthymemes. For the evaluation of statements during the dialectical process, four proof standards are used drawn from logic (dialectical validity) and from legal proof standards. Carneades Argument Graph have also been translated to the specification of Argument Interchange Format and vice versa [4].

**Argunet**\(^7\) [56] is a desktop tool coupled with an open source federation system for sharing argument maps. It is a multilingual environment which records the language of the map. An extensive online manual provides instruction, and they promote embedding debates. Users, have significant control over the presentation of arguments, such as colors and descriptions of different argument families. Finally, Argunet appears to support incremental formalization since arguments can be quickly sketched or reconstructed as premises and conclusions.

**Trellis**\(^8\) [10] is an interactive environment that allows users to add their observations, viewpoints, and conclusions as they analyze information by making semantic annotations to documents and other online resources. The system was built on Semantic Web technologies, inspired by intelligence analysis, began as a credibility and analysis system to help structure decisions. Trellis was designed to help capture argumentation, grounded in documents, whose reliability the user rated, and from which the user extracted statements. Argument can be extracted into XML, RDF, DAML, and OWL

---

\(^{5}\)http://discourcedb.org

\(^{6}\)http://www.convinceme.net

\(^{7}\)http://www.argunet.org

\(^{8}\)http://trellis.semanticweb.org
3.1. INTRODUCTION

3.1.2 Debating platforms

Several online platforms are implemented with the purpose of representing online debates and opinions in a more structured way. The objective of these systems is to motivate users participate by justifying their arguments, expressed as agreement or disagreement. Next, are following some of the most popular systems.

**e-Dialogos** [1] is a web application open public debate that enables citizens to connect with one another and discuss problems about Samos community for designing and implementing policies for the municipality. The system encourages citizens to participate and express their opinions that concerns city problems. Their approach is multi-disciplinary, combining elements of political science and communication, software and knowledge engineering to develop.

**CreateDebate**[9] is a social debate community. The platform has perspective debates, which generally have more that two sides, are scored based on user-applied tags. Some unique features of the system are that the debate moderator can add a topic research section with RSS feeds from other sites, offers numerous statistics for each debate such as the language level.

**Debate.org**[10] is a social networking site for debates. A unique feature of platform is that Debate.org’s focus on user profiles, allowing users to search for people with particular profile attributes such as income, location, gender, etc. Debates take place between two members and have four cycles: the challenging period, debating period, voting period and post voting period. Furthermore, determines the percentage to which other members agree with you on big discussions such as political hot topics.

**Debatewise**[11] is a web-based platform created with the aim to inform the users who like to make decisions on the basis of gathering information. In this platform, everyone can collaborate in creating the strongest case both for and against a given issue. As part of a partnership with iDebate, they provide links to Debatepedia and iDebate’s reference site Database. The system motivate users to participate in dialogues. There are several unique features such as editing the history of pro and con point. Debates are structured as adjudicated debates between two teams, other users can make comments, vote and subscribe to debates.

**iDebate**[12] is a debate platform where users can debate on a topic by providing statements for or against an initial position. Each statement consist of an argument and an counter-argument. There exist a set of voting types that users can select for voting at the beginning and end of debates but the platform does not support voting functionality on arguments or counter-arguments and the final debate results are calculated by vote aggregation on the two sides of initial position.

---

Deme [13] is a platform for online deliberation. Deme is designed to allow groups of people to engage in collaborative drafting, focused discussion, and decision making and specifically for supporting democratic, small to medium-sized group deliberation. This approach, however, does not capture much of the structural attributes of the arguments under discussions. While discussions and opinions may be identified by their topics, time, or participants there is a lack of fine-grained structure that captures how different facts, opinions, and arguments relate to one another and contribute to the overall picture.

LivingVote\textsuperscript{13}. Offers a debating platform that gives the advantage of voting procedure on each argument in a debate. There is the notion of supporting or attacking arguments to an initial statement and for each statement exist arguments that support or attack them. The platform contains features such as: adding their own arguments and the social support of arguments. As debate graph, this platform does not compute any debate outcome but only presents the accumulated votes that support or attack their initial arguments. Finally, the sub-graph votes are not considered.

3.1.3 Question and Answer websites (Q & A)

In this section, we will mention the most popular (Q & A) platforms that share similar functionalities with each other. The difference are also marked.

Quaestio-it\textsuperscript{14} [17] is a web-based Q & A debate platform that allow users to participate and express their comments and vote. It provides an interactive way for engaging into conversations regarding any question within the platform. The system describes a framework, based on computational argumentation for modeling and analyzing social discussions. Queastio-it uses an evaluation algorithm which highlighted the best comments.

Within platform, each answer is open for discussion so users can post their comments as supporting or attacking comments and vote. When posting an argument, users has to explicitly states the nature of argument relation (supporting or attacking argument). Some interesting features that debate platform support are the (i) browsing, where offers an interactive way for browsing through topics, (ii) debating, where discussions are presented in the form of trees, (iii) voting mechanism, where the system allows users to vote for a question, positively or negatively and finally (iv) logging mechanism, for users and social networks such as Facebook, Twitter and Google+.

StackExchange\textsuperscript{15} is a network of (Q & A) websites on topics in varied fields, each site covering a specific topic, where questions, answers, and users are subject to a reputation award process. The primary purpose of StackExchange site is to enable users to post questions and answer them. The social network supports several features such as voting mechanism for choosing the best answer, there is not

\textsuperscript{13}http://www.livingvote.org
\textsuperscript{14}http://www.quaestio-it.com
\textsuperscript{15}http://www.stackexchange.com
any restriction on commenting on answers only to one level which does not affect an answer’s rating. Users can vote on both answers and questions, and through this process users earn reputation points. The reputation point gives users more functionality on voting, which is not available for new users of the system.

**Quora** is an online collaborative information sharing and learning platform, that connects the information needs of real people with information solutions provided by real people. Some interesting features that the platform supports are the following: the user’s vote has different weight depending on the users activity. Answer are rating according to an algorithm based on votes, users profiles and spam/gaming detection techniques. Comparing to other platforms, Quora offers more complex discussions on commenting them. There exist an infinitive number of nested comments which do not contribute to an answer rating. Logging mechanism is also supported by the platform.

### 3.1.4 Visualization tools

Next, we describe the most popular web-based platforms for visualizing the different concepts of a dialogue.

**DebateGraph** is a web-based application that uses concept maps to explore topics and issues associated with them. It is a wiki debate visualization tool which have been adopted for use at the Keyoto climate change summit. It offers the opportunity to the public to collaborate on complex issues and evaluate all of the considerations that any member thinks may be relevant to the topic. The purpose of this platform is mostly to visualize the debates without providing any mechanism of how they reach in a debate conclusion. Furthermore, debategraph offers a way for somebody to learn about, deliberate and decide on complex issues. There is the notion of supporting and attacking arguments when participants involved in a conversation but there is not any evaluating mechanism for the arguments. Affords several visualizations, that can be embedded in other websites.

**ArgVis** is an Argument Visualization Tool designed to systemize the argumentation process by visually representing structured arguments to construct political deliberations. It aims to encourage people, especially the younger generation who are very familiar with new technologies, to express their opinions and contribute ideas to existing argumentations and be actively involved and participate in policy making. ArgVis structures argumentations in interactive graphs that comprise: Issues, Positions, Arguments and Counterarguments. The system supports the voting process, which offers the opportunity to select a preferred argument and express the agreement or disagreement with it. The framework of ArgVis can be separated into three main layers: the User Interface Layer (UI Layer), the Application Layer, and the Database layer. ArgVis’ argumentation model capitalizes on the components of a dialogue as defined by IBIS.

---

16 [http://www.quora.com](http://www.quora.com)
17 [http://debategraph.org](http://debategraph.org)
OnlineSpace\textsuperscript{18} is a software developed by UC Berkeley’s Center for New Media also designed to collect and visualize opinions on a variety of topics. Unlike Discourse DB, Online Space’s users are not afforded with a rich in its searching alternatives tool for opinions. Instead they are given the opportunity for real time deliberations in a way of declaring their agreement/disagreement with an opinion and stating their own positions regarding the topic. The notable point here is that they use sliders to express their sentiment to the specific opinion, entering the notion of strength on this interaction. The system then creates a map of opinions which is visualized as various sized points, each one representing a different perspective and with the larger points to map more popular perspectives. Neither the specific system uses a structure for the opinions and is mostly useful for visualization of real debates.

Cohere\textsuperscript{19} (Shum, 2008) \textsuperscript{[59]} is another Web-based argumentation tool that is intended to allow students and researchers to make personal and collective sense of problems. Users can create ideas and link them by means of different Connection Types. All connections are broadly classified as positive or negative. Cohere offers an attractive visualization that supports browsing and searching of an argument network.

SEAS\textsuperscript{20} \textsuperscript{[35]} \textsuperscript{[37]} is a template-based structured argumentation tool originally designed for collaborative intelligence analysis. Using this tool, contributing analysts directly manipulate depictions of arguments, adding and interpreting evidence relative to questions raised by the template, debate and draw conclusions based upon the collective evidence, and finally use these depictions to convey their findings to decision makers. SEAS’s most unique feature is its emphasis on templating, users can author templates which provide notions of how to make an argument, and specified authorized coeditors. The system visualization features are also considerable: to visualize multiple dimensions, SEAS uses starburst, constellation, and table views.

3.1.5 Synopsis

To sum up, we mentioned several software systems and described some of the most important features of every system. Next, we make a comparison between existing systems and our system in order to distinguish some of the similarities and differences among all works.

Various software systems are developed to assist users in analyzing textual arguments (e.g Araucaria; reed and Rowe, 2004), for helping users construct well-organized textual arguments (e.g Parmenides; Atkinson et al., 2006), and for organizing debates on the Web (e.g Cohere, Shum, 2008; Truthmapping, Debatemedia). However, little integration exist between these systems, mainly due to the lack of a unified ontology for describing argument structures. Our system uses an
3.1. INTRODUCTION

RDF ontology with the aim of making users aware of the conceptual components of a dialogue. Araucaria, for example, supports multiple argumentation schemes which allows switching between notations based on the Walton, Toulmin, and Wigmore models (Bex et al. 2003; wigmore 1931). Some of these systems focus on argument analysis and others on discussion analysis. By argument analysis, we mean the constructions of arguments or reconstruction of existing arguments. Comparing with other systems, our platform focus on analysing structured debates, considering well-explained opinions.

All systems encourage participants to be involved in structured debates and vote respectively but in a more strict way. For example, Debate.org allows the debate of only two participants while in StackExchange exists a restriction of commenting on answers only to one level. In our system, Apopsis, debates are not restricted on specific levels of commenting. Instead, they are increasing dynamically in any commenting level. Most of the systems can be applied successfully under specific conditions or domains. Our work is applicable in any domain.

Furthermore, considering all systems that we described above, we distinguish two of them that are close to our system, Apopsis. The most similar of them is the Quaestio-it.com system as we noticed that this system provides a reliable approach for evaluating the strength of each argument, an OWL ontology for representing the users opinions. Another system that is close to our platform is the e-Dialogos system which concentrates on enabling users to participate and vote on debates for deliberation purposes, a specific-problem domain where our system can be also applied successfully with the purpose of helping users understand the proposals and city planning policies expressed within discussions.

However, few of them provide a semantic ontology (OWL/RDF) for representing the users opinions or evaluation of the strength value of argument within debates but none of them support opinions analysis. The Apopsis system, applies a more reliable approach for evaluating the strength value of argument by aggregating both arguments and votes. Another important difference is that, most of the systems provide a debate with the purpose of coming up with an outcome while our system focus on displaying the different opinions as they are analyzed using clustering algorithms and associations rules and aims on making users understand how the dialogue is evolving when plethora of opinions and chaotic dialogues exist.

Finally, the Apopsis system supports semantic querying for retrieving the supporting and attacking arguments. Similar functionalities, we noticed in Avicenna and ArgDF systems, which they both provide semantic querying over a repository through an RDF/OWL ontology.

3.1.6 A taxonomy of the existing platforms

Next, we provide and compare some of the existing tools that are close to our system Apopsis according to specific features. For more comprehensive description of tools, see Appendix A.
### Table 3.1: Comparing existing tools (1)

<table>
<thead>
<tr>
<th>Domain</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Ontology-Theoretical model</th>
<th>Debating-support-attack relation</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quaestio-it</td>
<td>specific context, research problems and industrial settings</td>
<td>a platform used offering concrete solutions for both research problems and industrial settings</td>
<td>Extended Social Abstract Argumentation Framework (ESAAF) for evaluating the best answers</td>
<td>OWL ontology</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Debate.org</td>
<td>general domain</td>
<td>a social network for creating and participating in debates</td>
<td>Yes</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>iDebate</td>
<td>general</td>
<td>a platform for debating on a topic by providing statements for or against an initial position</td>
<td>set of voting types but not voting mechanism</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>CreateDebate</td>
<td>general</td>
<td>A social debate community for participating in debates</td>
<td>based on user-applied tags</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>StackExchange</td>
<td>several fields</td>
<td>The purpose is to enable users to post questions and answer them</td>
<td>Voting and Reputation mechanism</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Quora</td>
<td>general</td>
<td>An online collaborative information sharing and learning platform</td>
<td>Yes</td>
<td>no information provided</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>DebateGraph</td>
<td>public policy issues, collaboration</td>
<td>a platform that visualize the debates, collaboration on complex issues</td>
<td>No</td>
<td>IBIS model</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
### Table 3.2: Comparing existing tools

<table>
<thead>
<tr>
<th>Domain</th>
<th>Purpose</th>
<th>Voting-Score based system</th>
<th>Ontology-Theoretical model</th>
<th>Debating, support-attack relation</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parmenides</td>
<td>deliberative democracy, political domain</td>
<td>No</td>
<td>Douglas theory, argumentation schemes, computational argumentation</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>ConvinceMe</td>
<td>general</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Debatewise</td>
<td>general</td>
<td>Yes</td>
<td>no mechanism provided</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>DiscourseDB</td>
<td>news sources</td>
<td>No</td>
<td>No</td>
<td>do not provide online debates among users</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>LivingVote</td>
<td>general</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>e-Dialogos</td>
<td>deliberation domain</td>
<td>Yes</td>
<td>no available information</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Apopsis</td>
<td>general public problems</td>
<td>s-mDice, a framework for evaluating arguments</td>
<td>RDF representation, influenced from IBIS model used</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

3.1. INTRODUCTION
Chapter 4

Methodology

In this chapter, we discuss the system that we have implemented and mostly, the methodology for the analysis of various aspects of a dialogue, estimated with the help of clustering and associations rules.

4.1 Introduction

We have designed and implemented a Web-based System, called Apopsis, that supports argumentative discussions among participants that are evolved within dialogue. The major goal of the system is to summarize and capture various aspects of the dialogue analysis which allow the audience to have a better view of the ongoing debate.

Based on the users arguments, discussions are estimated with the help of clustering algorithms and association rules that are applied throughout the analysis decision-making process. These algorithms have been implemented and exist in WEKA¹, an open source software that contains a collection of machine learning algorithms for data mining tasks. The main clustering algorithms that are used in this work, consists of the K-means and Expectation-Maximization (EM) algorithms. While in associations rules, the system uses the most well known association rule mining algorithm, the Apriori algorithm.

Clustering Analysis is a data mining technique that group similar data objects into unknown groups of objects. The k-means algorithm aims at segmenting different items (users, arguments, positions) into groups with high intracluster similarity so that "similar" items (under some dimension, e.g., users with similar profiles, or users that agree with each other) will be assigned in the same group.

In order to determine which users share same views or profile characteristics with who, the K-means algorithm provides a distance measure similarity that aims to minimize the Sum of the Squared Error (SSE). One such distance measure, that is applied in our system is the the Euclidean distance. Another algorithm that is

¹http://www.cs.waikato.ac.nz/ml/weka/
applied in our methodology is the EM algorithm, which is an iterative probabilistic algorithm that generates probabilistic descriptions of the clusters. The goal we use this algorithm, is to generates the appropriate number of clusters according to the distribution and complexity of the data. The combination of these two algorithms provides a qualitative dialogue analysis considering all parameters that play a major role in this analysis.

Association rules are used for the users arguments analysis which discover the most similar and preferable positions that are expressed by a significant number of participants. Apriori is such an algorithm that we have used to make such analysis.

Furthermore, our system provides a range of functionalities that aims to motivate users to participate in structured dialogues, by expressing their opinions in form of arguments, positive or negative and vote. These functionalities include, (1) creating new debates, (2) a formal score function, called smDice, which evaluates the arguments, (3) and the ability of querying arguments from dialogues. We also provide a theoretical model of mapping the discussions within dialogue. The argument mapping schema consists of four elements, for instance, Issue, Position, Pro-argument and Con-argument. Moreover, we have used an ontology for describing and making users aware of the conceptual components of any domain. For that reason, common languages and principles are provided for querying arguments from an ongoing dialogue such as RDF, RDFs and SPARQL.

Finally, we describe the architecture of our system and every component in detail, which were important for implementing the Apopsis system.

4.2 Basic Concepts

Next, we introduce and describe the basic concepts of our implementation methodology in the system Apopsis.

4.2.1 Concepts Definition of our mapping schema

**Dialogue**: An exchange of ideas or opinions on a particular issue, especially a political or religious issue, with a view to reaching an amicable agreement or settlement.

**Structured Dialogue**: An instrument to ensure that opinions expressed within dialogues are well-justified.

**Semantic RDF Ontology domain**: Is an ontological formalization of the structural parts of a dialogue and we exploit it in the context of structured dialogues in order to support semantically queries on them such as retrieving supporting/attacking arguments.

**Issue**: An open debate topic where users are evolved with the goal of sharing their opinions in a positive or negative way through structured debates. An issue can be a question or a statement.
4.3 MAPPING SCHEMA

**Position**: A direct answer in form of statement, expressed by users who participate in a dialogue. Positions are the solutions or ideas to the Issue according to specific topics.

**Pro-argument**: A positive argument in response to a position or to another argument. The ability to support another opinion.

**Con-argument**: A negative argument in response to a position or to another argument. The ability to attack another opinion by providing well-justified arguments.

**Topic**: Represents labels or aspects of the issue matter. An argument is considered as an answer to specific topics on issue.

4.2.2 WEKA: A collection of ML Algorithms used in APOPSIS

**Clustering Analysis**: Clustering is the task of grouping a set of objects in such a way that objects in the same group (called cluster) are more similar to each other than to those in other groups (clusters).

**K-means algorithm**: K-means is one of the simplest and most commonly used unsupervised learning. Classifies a set of given data through a set of number of clusters.

**Expectation-Maximization (EM)**: The EM algorithm is an iterative probabilistic clustering algorithm that is used as a pre-processing procedure in our system and decide how many clusters to create by cross-validation.

**Apriori algorithm**: Apriori is an algorithm for frequent item set mining and association learning over transactional databases. Identifies frequent individual items in dataset and requires priory knowledge to generate the next generation of itemsets.

4.3 Mapping Schema

In this section, we will describe the various conceptual components of our work implemented in the web-based system, called Apopsis.

4.3.1 Theoretical model

Our system, Apopsis, provides argumentative dialogues where users are evolved by expressing their views in the form of arguments. The argument map, described below, is used for modeling discussions and the relations between them. In this part of work, we are influenced by the theoretical model, Issue-Based Information System (IBIS) [29] for describing the concepts of online dialogues that are implemented in our platform. We inherit some important features and represent some differences in regard with IBIS.
4.3.1.1 Comparison between IBIS and our mapping schema

Both mapping schemes, enhance the process of decision-making in communities. They are designed and meant to organize discussions, raised by problem-solving group. Our argument schema can be applicable in various domains while IBIS is especially intended to support community and mostly political decision-making. The argumentative element "issue" of each mapping schema differs on the fact that IBIS issue can be mostly a question, while in our mapping scheme, an issue can be either a question or a statement. Arguments are structured in defense of or against positions or other arguments. We separate the arguments into pro and con arguments. From our perspective, we define another concept, called topic, which play an important role when users come to response in issues or any other argument. Each argument (position, pro-argument or con-argument) will be a response to an issue according to a particular topic. Another difference is the different types of relations that exist between issues in the IBIS model in regard with our model which identifies a different type of relation, the "consistsOf" relation.

The basic concepts of our argument mapping schema are described below. The figure 2.1 shows how the argumentative elements are associated with each other by our work.

4.3.2 Semantic Web Ontology domain

Ontology is "the explicit specification of a conceptualization" (Gruper,1993) [20] that is used for sharing knowledge and making users aware of the conceptual components in specific domain. Despite the fact that more and more opinions are exchanged daily among participants on the Web, not many ontologies exist for describing the structure of these opinions. According to (Simsek et al., 2012), most of the ontologies have difficulties in modeling opinions due to the lack of
standards as how to model them or in lack of common interests between academic and commercials companies when it comes to extract opinions. Most of these ontologies describe concepts and features that it comes in specific domains. One such ontology is the Opinion Mining Core Ontology (OPM) proposed by Softic and Hausenblas (2008) [61], which describe the various concept of ontology in discussion with the connection of two existing vocabulary, the SIOC(Semantically Interlinked Online Communities) and the SKOS (Simple Knowledge Organisation System Reference). Our approach, represent an ontology that captures all the aspects of a dialogue that can be applicable in general domains.

Next, we are describing the basic concepts of our ontology in details, including all hierarchies (classes, subclasses and properties) that have arisen.

![Figure 4.2: Semantic Web Ontology.](image)

### 4.3.2.1 Describing the conceptual components of Semantic Web Ontology

Our mapping schema consists of 12 classes and 27 properties. Both entities, classes and properties begin with a unique identifier, consisting of a number proceded by the letter 'O' for classes and the letter 'P' for properties. Properties that start with the prefix (P) are the main properties of ontology that describe the relations among classes hierarchies while properties that start with the prefix (RP), are established with the goal of helping the querying procedure be faster than forming complex queries. Hence, all subclasses and subproperties that exist in our ontology schema inherit all features from the classes and the properties they belong to, respectively. Next, we are describing the conceptual components of the ontology in details. We start with describing the classes, the subclasses and how they are used by the system.
Classes and Subclasses hierarchies

**Creation Data**: This class constitutes a superclass of Issue, Argument and Topic classes. Contains information about the provenance of the object, for instance, the datetime and the person who created the object.

**Strength**: This class constitutes a superclass of Argument and Topic classes. Comprises information about the score calculation of arguments, for instance, comments, positions, votes etc.

**Content**: Constitutes a superclass of Issue, Argument and Topic classes. This class contains any information about the content of every object. Each content object has a type and a value.

**Issue**: Constitutes a subclass of Creation Data and Content classes which means that this class contains information about the content, the datetime and the person who created the issue. Is the main part of an argument schema as enable participants to be evolved in dialogues.

**Topic**: Constitutes a subclass of Creation Data, Content and Strength classes. A topic class can have a content, specific strength and a datetime when was created. Our system does not support a topic to have a strength, so far. Is closely related with the issue class, for instance, in the issue-question "Which smartphone is the best in the market today?", a topic can be specific label created by the user and constitutes of the following: "strongest battery, good performance, etc."

**Argument**: Constitutes a subclass of Creation Data class and a superclass of Position and Comment classes. This class represents a set of statements in the form of arguments, created by users which compose their arguments about one or more topics. Arguments can also be expressed in a positive or negative way which makes us distinct two different types of arguments, the pro-argument and the con-argument. An argument (positive or negative) can be support or attack a position or other arguments.

**Position**: Constitutes a subclass of Argument and inherit each characteristics from all above superclasses. A dialogue contains one or more positions. As an argument, comprises information about the content, the datetime that was created and the person who created the position. Every position is closely related to topics when it comes to be composed.

**Comment**: Constitutes a subclass of Argument and inherit each characteristics from all above superclasses. A dialogue contains one or more comments which can be positive arguments or negative arguments. A positive argument constitutes a positive reply to that argument and negative argument constitutes a negative reply where users need to justify their arguments in favor of various aspects that the system support. In more particular, when a negative arguments is formed, the user is obligated to choose among the aspect (*Incorrect, Irrelevant, Insufficient*), in order to reply to that argument.

**Rating**: Constitutes a superclass of Vote and Star-Based classes. This class
comprises information about the general rating of the dialogue, including the comments, position, etc. The rating system supports various forms such as 5-stars system, positive/negative arguments, etc. Our system supports the voting mechanism only. Hence, the voting mechanism refers to votes that can be either positive, or negative upon position and comments.

**Category:** Constitutes a subclass of Creation Data class. This class organizes the issues into categories which is implemented manually by allowing the user to select the type of category when he wants to open a debate issue.

**Person:** This class constitutes an already existing ontology that comprises information about the users profile characteristics. In our implementation, we have used 11 profile characteristics of the Person\(^2\) ontology, which consists of the following elements: email, name, accessCode, gender, birthDate, givenName, familyName, alternateName, Address, has education and marital status. The attributes has education and marital status do not exist in the Person ontology and have been extended by our system in order to use them in the dialogue analysis.

**Properties hierarchies**

**has-content-value:** This property identifies the value of the content that an argument can have. The content value can be a text, a video, an image, music track, etc. In our system, we use the text form of content value. We assume that a dialogue contain arguments in form of text content.

**has-content-type:** This property is mentioned to the type of the content that an argument can have. We identify 4 different types which are, TEXT, IMAGE, VIDEO, EMPTY.

**datetime:** This property is mentioned to the datetime of any argument, topic or position that are created by specific user in the system.

**voting:** This property identifies the value of strength that arguments and topics obtain. In our case, strength have the arguments only (position, comments).

**dialogue:** This property identifies the value of dialogue strength that arguments and topics obtain. Is used for calculating the acceptance and quality of the arguments.

**congruence:** This property identifies the value of acceptance score that arguments and topics obtain. The acceptance is refers to how acceptable an argument is, depending on the numbers of comments that support or attack an argument.

**quality:** This property identifies the quality score of an argument or topic. Quality score is refers to how well-justified the argument is according to the user’s answer.

**created by:** This property identifies the user who created the argument or topic in a dialogue for one or more topics.

**consist of:** This property identifies the topics that an issue contains when it is created by the user. As a relation, connect the issue class with existing topics.

\(^2\)http://schema.org/Person/
about topic: This property identifies the topics that are connected with arguments that exist in the system. It is a useful property when we want to bring all arguments that are related with specific topics in a dialogue.

attacks: An important property that represents the relation between arguments, denoting that one argument attack/disagree another argument.

supports: An important property that represents the relation between arguments, denoting that one argument support/agree another argument.

replies: A property that identifies a simple relation between two arguments. This property is not supported by our system.

related to position: A property that connect every argument which exist in the system with a specific position. This is a redundant property that is helping in cases where we want to get all arguments that are related with specific position.

rates: A property that connect every argument with a voting. A voting can be positive(+1) or negative(-1).

contains: This property is refers to the votes that an argument contains either positive, or negative. Connect every vote with specific argument.

classistics of: This property is not supported by the system.

has rating: This property identifies the rating value. For example, when a user votes up for a comment(position or argument) the vote value is recorded as +1 in the system, otherwise the vote value is recorded -1 by voting down the comment.

status: This property identifies the status of every comment(position or argument). The status contains 3 different values such as ACTIVE, OUTDATED and DELETED. Our system, identifies a status value for an issue as well such as CLOSED. It is used when a debate issue is closed by the moderator.

reviews: This property describes the views that each issue have received according to how many times an author visit or not a debate conversation.

belongs to: This property is associated with the different stages that are implemented in our system. Identifies which positions belong to the first stage and which are transferred to the second stage of discussion.

closed time: This property represent the time of closing a debate issue when there is no need of keeping it active. This action is supported by the moderator of the system who has created that issue.

related to: This property describes the association of any position with an Issue according to specific topics. An issue can have one or more positions.

4.4 Our approach

This section describes the methodology that we propose for the analysis of various aspects of a dialogue. We will mention the set of algorithms that are used and how they are combined in order to cover the different information needs. Also, we will make a reference to the score function that is used by the system with the goal of evaluating the strength of arguments. Finally, we will mention all information needs that we have used in our debate analysis, in details.
4.4. OUR APPROACH

4.4.1 s-mDiCE: A formal framework for evaluating arguments

Our system, Apopsis uses a formal framework for evaluating the strength of arguments, called s-mDiCE [44], which is generic enough to capture the features of a variety of domains, such as debate portals and decision-making systems. This framework combines features, such as votes or expert rating, supporting and attacking arguments. The evaluation of arguments is distinguished between the two values of score, the acceptance of argument which represent, how acceptable an argument is, based on the number of comments that support or attack that argument and the quality of argument that represent, how well-justified a user-generated comment is presented. Aspects allow accomplishing the distinction between the two scores.

In this work, we use the score function in order to evaluate the strength of arguments that have been expressed by users in a form of supporting or attacking arguments, considering one or more aspects. Each argument score starts with a base score (0.5), which is used to capture an expert’s initial rating over an opinion, before any debate has taken place. Depending on the type of arguments, the score is increasing or decreasing. The figure 4.3 describe how the concepts of a debate graph, such as the arguments and votes are associated with each other in order to provide a score value for the user-generated arguments.

Figure 4.3: (a) A debate graph with votes, base score and user-generated supporting and attacking arguments, (b) its transformation with blank nodes.

**figure (a)**: We consider an argument where a set of arguments are reacting on that either positively, or negatively and the strength value of argument is changing in a positive or negative way, according to the confidence and reliability that arguments provide. The same is applied for votes. A positive vote indicates an increasing value on argument, a negative vote indicates a decreasing value on that argument.

**figure (b)**: We assume the same previous graph, further comprising the blank argument metaphor which carries no content of its own, rather it shares the content and rationale of the target argument. In case no debate have taken before, the strength value of each argument is calculated in the same way as the figure(1)
shows. When already exist a debate, the strength value of an argument is calculated, considering the hypothetical argument blank argument. Reacting on an argument 'a' positively, it means that you react either positive to the blank node with embedded positive arguments/votes or negative to the blank argument with embedded negative argument/votes. Respectively, reacting on an argument negatively, it means that you react either positive to the blank node with embedded negative arguments/votes or negative to the blank node with embedded positive arguments/votes.

4.4.2 Opinions Analysis

Our work, aims at analyzing the various aspects of an ongoing dialogue, considering not only the user-generated arguments but also the user profile characteristics. The proposed analysis, covers different information needs, emerging from users in order to reduce the burden of having to go through entire debate.

A collection of Machine Learning algorithms have been used for analyzing discussions, such as Clustering algorithms and Associations Rules. We use K-means for determining similar users or similar positions and Apriori for finding interesting relationships between those users or positions. We also apply different combinations of these two algorithms, according to the information needs that exist for opinions analysis. Next, we describe all parameters that are used for each algorithm throughout the analysis decision-making process and we provide all "Best Practices", in details.

4.4.2.1 Describing algorithms

Next, we mention each one of the algorithms and the parameters that are used in opinions analysis such as K-means, Expectation-Maximization (EM) and the Apriori algorithms.

K-means algorithm: K-means algorithm constitutes the most important algorithm in our cluster analysis implementation which groups similar objects with high intrACLuster similarity(users, positions and similar profiles) that are identified within dialogues in the system Apopsis. In order to determine the similarity of objects and how these objects converge in the same cluster, K-means uses a distance similarity measure, called Euclidian Distance Measure. The number of clusters is determined with two different ways. Firstly, we run multiple times the algorithm with different parameters each time in order to get the minimum sum of squared error (SSE). Secondly, we used another clustering algorithm, the Expectation-Maximization (EM) algorithm in order to define the appropriate number of clusters according to the data. During the K-means process, we applied many runnings with different values of iterations and the most appropriate and valid running according to the dataset we provided, was the iteration (I=500). For
the initial centroids, we used the value of seed \( S=100 \). Finally, in our implementation, we do not replace any missing value as the missing values represent unknown answers from users who participate within a dialogue.

**Expectation-Maximization (EM) algorithm**: The EM algorithm is used as a pre-processing procedure in K-means algorithm in order to determine the appropriate number of clusters in order to perform cluster analysis. The input of EM algorithm is the number of clusters that we need to set equal to \((-1)\) which means that the algorithm will perform cross-validation process for calculating the number of clusters. The number of iterations that algorithm perform is equal to \((I=500)\). Finally, the initial centroids we used is equal to \((S=100)\).

**Apriori algorithm**: Another important algorithm we use in opinions analysis is the Apriori algorithm which determines interesting relationship between positions and users profiles within dialogues. In this algorithm, we need to determine the number of rules that will be produced during the running procedure which are equal to \((N=5)\). An important parameter is that of metric type where we define the confidence as the type of measuring the reliability of inference made by a rule. Default values of parameters are the \(D=0.05\), the lower bound equal to \((M=0.1)\) and the upper bound equal to \((U=1.0)\).

<table>
<thead>
<tr>
<th>K-means Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Definition</strong></td>
<td>The K-means algorithm is an unsupervised learning algorithm, which determines similar object within clusters and dissimilar object out of clusters.</td>
</tr>
</tbody>
</table>
| **Number of Clusters** (\(N\)) | 1. Defined by the Expectation Maximization Algorithm (EM).  
                            2. Multiple running until you get the minimum sum of squared error (SSE), known as Elbow method. |
| **Max-Iterations** (\(I=50,100, 500\)) | Represent the max iterations that are applied during the K-means analysis process. |
| **Distance measure** | The algorithm uses the Euclidian Distance Measure mostly, which represents the degree of correspondence among object across all of the characteristics that exist in data. Other distance measure that has been applied, is the Manhattan distance. |
| **Seed** (\(S=10, 50, 100, 500\)) | Determine the initial centroids of clusters of the K-means algorithm that are produced randomly. |
| **Missing Values**   | We do not replace the missing values in K-means algorithm. |

Table 4.1: K-means parameters
## CHAPTER 4. METHODOLOGY

### EM Parameters

<table>
<thead>
<tr>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The Expectation Maximization Algorithm (EM)</strong> is used as a pre-processing algorithm in K-means algorithm. Determines the number of clusters that need to be used.</td>
<td></td>
</tr>
</tbody>
</table>

| **Number of Clusters (N)** | The number of cluster is set to (-1) in order Cross-Validation to be performed and then determines the clusters. WEKA supports Cross-Validation by setting the N=-1. |

| **Max-Iterations:** (I=50, 100, 500) | Represent the max iterations that are applied during the EM running process. |

| **Seed (S=10, 50, 100, 500)** | Determine the initial centroids of clusters of the K-means algorithm that are produced randomly. |

| **Table 4.2: EM parameters** |

### Apriori Parameters

<table>
<thead>
<tr>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Apriori algorithm is an association rule mining algorithm that determines interesting relationships between data attributes on a huge dataset.</strong></td>
<td></td>
</tr>
</tbody>
</table>

| **Number of Rules (N=2, 5, 10, 20)** | The number of rules that need to be determined during the Apriori running process. |

| **Delta (D=0.05)** | Represent a factor which iteratively decrease the support. The support is reduced until the min-support has reached or until the number of rules has been generated. |

| **Lower bound of min-Support (M=0.1)** | Support is an indication of how frequently the item-set appears in our data. This parameter is mentioned to the lower bound of the minimum support. |

| **Upper bound of min-Support (U=1.0)** | It is mentioned to the maximum bound of min-support. |

| **Metric type (C=0.9)** | There are various metrics types for measuring the reliability of inference made by a rule, such as i) Confidence, ii) Lift, iii) Leverage and iv) Conviction. In our case, we use the confidence metric for determining the inferences rules. Confidence is the conditional probability that the consequent will occur given the occurrence of the antecedent. |

| **Table 4.3: Apriori parameters** |
4.4. OUR APPROACH

4.4.2.2 Describing the "Best Practices"

By the term "Best Practices" we mean the various types of debate analysis, emerging from users during an ongoing dialogue. The system has been implemented 6 different types of information needs. We will describe each one of the information need, in detail. Description provides more information about the information needs we want to make the analysis. Requirements are referred to the elements of an ongoing debate that need to be filled by the user throughout the analysis decision-making process. Proposed method is referred to the methodology that we use, including the algorithms we explained earlier, in order to have the preferred analysis results.

- Users with similar preferences

<table>
<thead>
<tr>
<th>Information need</th>
<th>Users with similar preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>We need to find groups of users, who share same opinions with a set of positions, which are related with one or more topics. Hence, we determine the user profile characteristics of every group of users that arises from the debate analysis.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Issue, Topics, Positions</td>
</tr>
</tbody>
</table>

Table 4.4: Users with similar preferences

Proposed method
This type of analysis consists of 3 different procedures: the filtering mechanism, the K-means algorithm and the Apriori algorithm.

At first, we use a filtering mechanism which allow users to select among issue, topics and positions according to the system interface as it is designed. Users can select one issue each time and one or more topics and positions. Next, we use the clustering algorithm and associations rules in order to determine the different groups of users and their similar preferences either positions or profile characteristics.

Specifically, we use the K-means algorithm to determine the different clusters that were created during the algorithm execution. For each cluster that was created, we apply Apriori algorithm in two cases: In the first case, we want to find the positions that users have in common, in order to determine which users share same opinions with whom, and second we use apriori to determine the similar profile characteristics among these users that belong in the same group. It is not necessary that users who were partitioned by the clustering procedure and have similar views, will also share same characteristics. The groups of users may share similar opinions or similar profile characteristics or both of them.
### Chapter 4. Methodology

#### Algorithms

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>authors and positions</td>
<td>1. each cluster of users generated from K-means and positions. 2. each cluster of users generated from K-means algorithm and profiles.</td>
</tr>
<tr>
<td>Output</td>
<td>clusters of users</td>
<td>1. positions 2. profile characteristics</td>
</tr>
</tbody>
</table>

Table 4.5: Information need (1) : Input/Output

- **Sharing similar views with others**

<table>
<thead>
<tr>
<th>Information need</th>
<th>Sharing similar views with others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>We need to find groups of users according to username whose opinions are close to those expressed by a set of users, which are related to one or more topics.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Issue, Topics, Authors</td>
</tr>
</tbody>
</table>

Table 4.6: Sharing similar views with others

Proposed method

This type of analysis consists of 3 different procedures: the **filtering mechanism**, the **K-means algorithm** and the **Apriori algorithm**.

At first, we use a **filtering mechanism** which allow users to select among issue, topics and authors according to the system interface as it is designed. Users can select one issue each time and one or more topics and authors. Next, we use the clustering algorithm and associations rules in order to determine the different groups of users that share same opinion with other users.

Specifically, we use the **Apriori algorithm** to determine the common positions they share with authors, who have chosen from the filtering mechanism process. For each collection of positions they have in common, we apply **K-means algorithm** to determine the groups of users that share same opinions with others. Finally, we define for each cluster, the users names by making the appropriate SPARQL query in the RDF repository, the Virtuoso.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>each cluster of positions generated from Apriori algorithm and authors</td>
<td>users and positions</td>
</tr>
<tr>
<td>Output</td>
<td>users</td>
<td>clusters of positions</td>
</tr>
</tbody>
</table>

Table 4.5: Information need (1) : Input/Output
4.4. OUR APPROACH

Table 4.7: Information need (2): Input/Output

- Same users with similar views

<table>
<thead>
<tr>
<th>Information need</th>
<th>Same users with similar views</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>We need to find groups of users with similar opinions with a set of positions who share specific profile characteristics according to one or more topics.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Issue, Topics, Authors Profile, Positions</td>
</tr>
</tbody>
</table>

Table 4.8: Same users with similar views

Proposed method
This type of analysis consists of 3 different procedures: the filtering mechanism, the K-means algorithm and the Apriori algorithm.

At first, we use a filtering mechanism which allow users to select among issue, topics and authors profile according to the system interface as it is designed. Users can select one issue each time and one or more topics and authors profile. Next, we use the clustering algorithm and associations rules in order to determine the same groups of users who share same opinion with specific set of positions. Specifically, we use the K-means algorithm to determine the users that satisfy specific profile characteristics as they are chosen from specific user through the system interface. For each set of users, we apply the Apriori algorithm to define the similar views.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>users and profiles</td>
<td>each cluster generated from K-means algorithm and positions</td>
</tr>
<tr>
<td>Output</td>
<td>clusters of users</td>
<td>positions</td>
</tr>
</tbody>
</table>

Table 4.9: Information need (3): Input/Output

- Users with different views

<table>
<thead>
<tr>
<th>Information need</th>
<th>Users with different views</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>We need to find groups of users who agree or disagree on set of positions that have received specific score.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Issue, Topics, Scores, Positions</td>
</tr>
</tbody>
</table>
Proposed method
This type of analysis consists of 3 different procedures: the filtering mechanism, the K-means algorithm and the Apriori algorithm.
At first, we use a filtering mechanism which allow users to select among issue, topics, scores and positions according to the system interface as it is designed. Users can select one issue each time and one or more topics, scores and positions. Next, we use the clustering algorithm and associations rules in order to determine groups of users that agree or disagree with specific set of positions. Specifically, we use the K-means algorithm to determine different clusters of users that share similar views. Next, for each one of the clusters, we apply the Apriori algorithm to define the positions that groups of users have in common within cluster.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>users and positions</td>
<td>each cluster generated from K-means algorithm and positions</td>
</tr>
<tr>
<td>Output</td>
<td>clusters of users</td>
<td>positions</td>
</tr>
</tbody>
</table>

Table 4.11: Information need (4) : Input/Output

• Positions according to similar preferences

<table>
<thead>
<tr>
<th>Information need</th>
<th>Positions according to similar preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>We need to find groups of positions which are expressed by users with similar preferences and profile characteristics according to one or more topics.</td>
</tr>
<tr>
<td>Requirements</td>
<td>Issue, Topics, Authors Profile</td>
</tr>
</tbody>
</table>

Table 4.12: Positions according to similar preferences

Proposed method
This type of analysis consists of 3 different procedures: the filtering mechanism, the K-means algorithm and the Apriori algorithm.
At first, we use a filtering mechanism which allow users to select among issue, topics and authors profile according to the system interface as it is designed. Users can select one issue each time and one or more topics, authors profile. Next, we use the clustering algorithm and associations rules in order to determine groups of positions that have been expressed by users who satisfy similar profile characteristics.
4.4. OUR APPROACH

Specifically, we use the K-means algorithm to determine different clusters of users according to the profile characteristics that have been chosen. Next, for each one of the clusters, we apply the Apriori algorithm to define the positions that are expressed by specific users in the system. As result, we get positions as they are formed by users in dialogues.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
<td>users and profile</td>
<td>each cluster generated from K-means algorithm and positions</td>
</tr>
<tr>
<td><strong>Output</strong></td>
<td>clusters of users</td>
<td>positions</td>
</tr>
</tbody>
</table>

Table 4.13: Information need (5) : Input/Output

- Topics based on users views

<table>
<thead>
<tr>
<th>Information need</th>
<th>Topics based on users views</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>We need to find groups of topics according to users agreement/disagreement on positions, which are related to specific issue.</td>
</tr>
<tr>
<td><strong>Requirements</strong></td>
<td>Issue, Authors</td>
</tr>
</tbody>
</table>

Table 4.14: Topics based on users views

Proposed method

This type of analysis consists of 3 different procedures: the filtering mechanism, the K-means algorithm and the Apriori algorithm. At first, we use a filtering mechanism which allow users to select among issue and authors according to the system interface as it is designed. Users can select one issue each time and one or more authors. Next, we use the clustering algorithm and associations rules in order to determine groups of topics that exist in the system and are related with users opinion either they agree or disagree on these topics.

Specifically, we use the K-means algorithm to determine different clusters of users according to their answers(positions) either positive or negative. Next, for each one of the clusters, we apply the Apriori algorithm to define those positions, so we can get then the topics that are related with these positions. As result, we get groups of topics that are closely related with specific user’s answers.
### Table 4.15: Information need (6) : Input/Output

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>K-means</th>
<th>Apriori</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input</strong></td>
<td>positions and users</td>
<td>each cluster generated from K-means algorithm and positions</td>
</tr>
<tr>
<td><strong>Output</strong></td>
<td>clusters of users</td>
<td>topics</td>
</tr>
</tbody>
</table>
Chapter 5

Implementation

In this section, we discuss the system that we have implemented, the functionalities of the system provide and the basic architecture of Apopsis.

5.1 Apopsis: A debating platform

Apopsis (http://www.ics.forth.gr/isl/Apopsis) is a web-based debating platform that allow users to open new topics of discussions, raise issues, post comments and vote. As a debate portal, motivate people to participate in structured, goal-oriented dialogues, where users can post their comments, expressing their agreement and disagreement to the answer respectively. Debates are presented as trees and each one is created by the users of the system to initiate a conversation about a goal-oriented topic. Based on users arguments, discussions are estimated and evaluated through an evaluation algorithm, called s-mDice. A Semantic Web Ontology is used for modeling and representing the users opinions in a structured way.

The main objective of the platform is to provide different means of analysis of debates, in order for participants to get a better understanding of the validity and justification of each individual opinion expressed. The debate analysis is not directed to a response of yes or no. Instead, the decision-making process generates different groups of users either according to their beliefs or to the profile characteristics, who share or not similar opinions with other participants on a variety of topics. For the analysis purpose, we implemented 6 different information needs, emerging from users, engaged into conversations throughout the analysis decision-making process. Finally, the system performs semantic querying for retrieving the supporting and attacking arguments through SPARQL queries.
5.2 Functionality

We have designed and implemented a Web-based system, called Apopsis, which motivate users to participate and collaborate in dialogues, organize discussions in a structured way and the most important feature of this system is to analyze various aspects of dialogues, providing different means of the analysis of opinions expressed in debates. Furthermore, the system provides a range of functionalities, which are mentioned below:

- An RDF ontology, for representing the structure of knowledge domains that stores and retrieves discussions in form of RDF statements.
- A formal framework, called sm-Dice, for evaluating the strength value of each argument.
- A combination of ML algorithms, for the clustering analysis of users opinions, including clustering algorithms for the clustering of features and associations rules for the extraction of interesting rules.
- Logging mechanism for actively participating in debates.
- Creating new topics of discussion in order to encourage users to participate in debates.
- Creating new position/arguments of ongoing dialogues.
- Vote up/down upon positions/arguments.
- Semantic Querying for retrieving the supporting and attacking arguments
5.2. FUNCTIONALITY

- Less important features are considered the following view lists: *view list of users, view list of moderators and view list of topics* created throughout the discussion process.

- Two different phases: debates proceed in two different stages, which are facilitated by a class of users, known as ”moderators”, ensuring the high and quality analysis of opinions.

5.2.1 Logging mechanism

Apopsis, provides a logging mechanism that allow users to register into the system and then participate and collaborate with other contributors within debates by posting arguments and vote upon existing arguments. The logging mechanism is making more accurate the clustering analysis of opinions as we consider the profile characteristics of every user in our analysis decision-making process. The profile characteristics that the system supports for registration are mentioned below: *firstname, lastname, username, email, password, age, marital status, gender, education and user image.*

<table>
<thead>
<tr>
<th>Profile Characteristics</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>18-25 , 26-46 , 47 and more</td>
</tr>
<tr>
<td>Marital Status</td>
<td>Single , Married , Divorced</td>
</tr>
<tr>
<td>Gender</td>
<td>Male , Female</td>
</tr>
<tr>
<td>Education</td>
<td>Primary School , High School , University</td>
</tr>
</tbody>
</table>

Table 5.1: User Profile Characteristics

5.2.2 Creating new debates

New debates can be created by users who wish to participate in debates for a decision-making problem. The system provides the ability of organizing debates.
into different categories where users can navigate among different dialogues by either choosing the topic of discussion that wish to participate or creating new debates. For creating new debates, users need to construct the topic of discussion, called issue and then choose appropriate topics that are closely related to the topic of discussion. The organization of debates is performed by the user by choosing among different categories, the most relevant with the issue. Next, we are showing the creating new debate form.

![Create New Debate Form](image)

Figure 5.4: Creating new debate form.

### 5.2.3 Creating new positions

Every user that wish to participate in dialogues, they can create one or more positions in order to express an idea or their beliefs in the context of an ongoing dialogue. Next, we are showing two figures, (1) the creating new position form which contains two different topics (decision-making process and helping) and (2) a position argument, showing a position as it is displayed in the system during the conversation part of a dialogue.

![Create New Position Form](image)

Figure 5.5: Creating new position form.

![Position Argument](image)

Figure 5.6: A position argument.

### 5.2.4 Creating new arguments: supporting/attacking arguments

Users can create new arguments through the system interface in order to support or attack either positions or arguments. They can create one or more arguments within dialogue as the system provides multilevel of comments. In Apopsis,
debates are presented as trees where the root corresponds to the issue, its immediate children to the positions and all other subsequent level nodes correspond to the arguments either positive, or negative arguments. Users can also vote upon these arguments with a positive or negative way according to their beliefs.

![Figure 5.7: Creating new argument.](image)

![Figure 5.8: Supporting/attacking arguments.](image)

### 5.2.5 Vote up/down on positions/arguments

In the system, users can vote up/down on positions or arguments according to specific aspects that are supported by the system. Apopsis helps participants justify their agreement/disagreement or votes according to the following aspects: Incorrect, Irrelevant and Insufficient.

![Figure 5.9: Vote on positions/arguments.](image)

### 5.2.6 Searching mechanism: retrieving supporting/attacking arguments

Another functionality of our system is the searching mechanism for retrieving arguments (supporting/attacking) that have been expressed within debates. Requirements are referred to the fields that must be filled by users for searching purpose. We have implemented 6 different types of searching and we illustrate them in the figure 3.5.
CHAPTER 5. IMPLEMENTATION

Figure 5.10: Types of searching mechanism.

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for arguments that can support/attack positions or arguments which are associated with specific aspects.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of relation, Aspects, Positions, Arguments</td>
</tr>
</tbody>
</table>

Table 5.2: Searching type 1

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for arguments that are supported or attacked by other arguments which are associated with specific aspects.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of relation, Aspects, Arguments</td>
</tr>
</tbody>
</table>

Table 5.3: Searching type 2

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for arguments or positions that have received specific score.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of arguments, Scores</td>
</tr>
</tbody>
</table>

Table 5.4: Searching type 3

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for arguments or positions that are posted by specific author.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of arguments, Authors</td>
</tr>
</tbody>
</table>

Table 5.5: Searching type 4

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for arguments or positions that have been evaluated as the best arguments according to specific issue.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of arguments, Issues</td>
</tr>
</tbody>
</table>

Table 5.4: Searching type 3
Table 5.6: Searching type 5

<table>
<thead>
<tr>
<th>Description</th>
<th>Search for authors whose arguments have received the best score, according to specific issue.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements</td>
<td>Type of arguments, Issues</td>
</tr>
</tbody>
</table>

Table 5.7: Searching type 6

5.2.7 Two phases of dialogues

In Apopsis, dialogues proceed in two phases: The first one includes an open dialogue where users share their opinions and ideas on specific issue; the positions that attract significant strength by the participants, progress to the second phase by a moderator, who is charged with the task of supporting a more focused, high quality dialogue.

5.3 Technical Information and Architecture

The Web-based system, APOPSIS, is available in http://www.ics.forth.gr/isl/apopsis. The implementation is developed as a web application project. It is written mainly in Java and JavaScript, while we used other web development technologies such as HTML5, CSS3, JSP, jQuery, JSON and AJAX requests. The system is developed in the platform Netbeans and the server we used is the glassfish default server of Netbeans. Also, we used a Bootstrap\(^1\) in order to make the front-end web development faster and easier. Finally, APOPSIS consists of three main tiers: the Client Tier, the Server Tier and the Data Tier which are illustrated in figure (5.11).

**Server Tier**: Considering the backend of our system, we designed a system which undertakes to analyze opinions considering different aspects of a dialogue. The server-side part is a Java application that was developed in order to provide a complete API of all provided functionalities. It was designed as a servlet Web-based Service, using Java API and Semantic Web Standards such as RDF statements for representing the conceptual components of a dialogue. For the purpose of analyzing debates, we used WEKA, a collection of ML algorithms that combine clustering algorithms and associations rules.

**Client Tier**: The client-side environment constitutes the user interface of our system. It is implemented based on modern design patterns, following the basic user interface design principles. The client-side environment consumes the servers services as JSON data and provides representations using all the latest Web features like JSP, HTML5, CSS3, Ajax requests and JQuery that all modern browsers

\(^1\)http://colorlib.com/polygon/gentelella
support. The Ajax Requests work as intermediary between client and server, orchestrating data exchanges. It directs the data in order to send and receive the appropriate contents, making asynchronous requests to server API.

**Data Tier:** The data-tier represents a RDF repository, Virtuoso, which is used for storing and retrieving arguments in the form of RDF statements with the help of SPARQL queries. For the purpose of facilitating opinions, the system uses an RDF ontology that represent arguments and their relations in order to make users aware of the conceptual components of an ongoing dialogue. Next, we illustrate the basic system architecture of Apopsis.

![Figure 5.11: Basic architecture of Apopsis.](image)

### 5.3.1 Openlink Virtuoso Universal Server

OpenLink Virtuoso Universal Server is a cross-platform universal server that acts as a virtual database engine that combines the functionality of a traditional RDBMS, ORDBMS, virtual database, RDF, XML, free-text, web application server and file server functionality in a single system. The software has been developed by OpenLink Software with Kingsley Uyi Idehen and Orri Erling as the chief software architects. The key features of OpenLink Virtuoso Universal Server are summarized on the figure 3.1.

Apopsis was developed over jdbc Virtuoso provider, allowing users of Virtuoso to leverage the jdbc framework to modify, query, and reason with the Virtuoso quad store using Java language. For the needs of Apopsis jdbc is used as a Java library, providing tools to parse, interpret, query and store this information, embedded in Apopsis.
Chapter 6

A System Use Case

6.1 Main features of APOPSIS platform

Next, we introduce a simple system use case in order to describe the main features of the apopsis system. For that reason we have created debates, based on dialogues that exist in the platform debate.org. We follow a step-by-step procedure, explaining the functionalities of the system and showing how we can interact with it. Finally, we provide results for each functionality supported by the system.

6.1.1 Two-stages of dialogue

Debates proceed in two different stages; the first stage contains a free conversation where users can contribute with each other by commenting in the form of supporting or attacking opinions or voting on arguments, while in the second stage of debate, the system applies opinions analysis, considering arguments with a high score and user profile characteristics. In the next figure, we show how specific positions proceed to the next stage of conversation according to users selection. Let’s consider, the user choose to transfer 4 of the 6 positions that existed in the debate.

![Figure 6.1: An instance of creating a new debate.](image-url)
6.1.2 Creating new debates

In the first interaction with the system, user can navigate among existing debates or choose to create a new debate. Let’s consider a user\(^1\) who choose to *create a new debate*, then the user need to fill all fields in order to initiate a conversation concerning a particular issue related to the question: “*Do violent video games cause behaviour problems and how?*”.

- **Category**: Technology
- **Issue**: *Do violent video games cause behaviour problems and how?*
- **Topics**: increase violence, escaping from reality, corrupts teenagers minds, stress relievers

![Create New Debate](image)

Figure 6.2: An instance of creating a new debate.

6.1.3 Creating new positions/arguments

After creating a new debate, users can participate in conversations and contribute with each other by commenting either positively or negatively, or creating new ideas (positions) on the topic discussion. Next figures show 1) an instance of creating new position and 2) another instance of creating new argument.

\(^1\)Each user in the system need to register or login first before interact with the system APOPSIS.
6.1. MAIN FEATURES OF APOPSIS PLATFORM

6.1.4 Evaluating the strength value

The system applies an evaluation method (sm-Dice) that calculates the strength value of each argument by aggregating the arguments and the votes upon arguments. Aposis incorporate the concept of "aspects" which are associated with; how well-justified the user answers stand. For now, the platform includes three different aspects: Incorrect, Irrelevant and Insufficient.

6.1.5 Opinions Analysis

The system applies opinion analysis in order to 1) identify the most relevant opinions, 2) display the different opinions expressed within dialogues and 3) finding the common characteristics among participants. We implemented six different information needs, emerging from users during the decision-making procedure. Next, we perform dialogue analysis by providing instances for each one of the information needs implemented in the system.

Clustering Query: "Find groups of users who share the same opinions with a set of positions, which are related with one or more topics. What are the profile
characteristics they have in common?”.

**Description of the query execution**: In this execution, the author has chosen to find groups of users who share same views with the selected positions which are related to specific topics as it is shown from the screenshots. If there are not any results, the system return "No debate analysis results were found. Please try again!” for this searching applied.

**Analysis Results**

**Description of the opinions analysis results**: According to the results, there were found 3 different groups where:

- The 1st group share same opinions with the following positions *Studies have shown if you play violent video games you have bad behavior* and *They are
6.1. MAIN FEATURES OF APOPSIS PLATFORM

detrimental to teens minds and society. This group also shares same profile characteristics where users within group are male and divorced.

• The 2nd group share same opinions with the following three positions Studies have shown if you play violent video games you have bad behavior, They are detrimental to teens minds and society and In my experience video games do not increase violence. They do more good than bad. Furthermore, in this group we distinguish users where they can share same opinions but they do not share same profile characteristics. All of them have the same gender (male) but they differ in their age, education or marital status.

• The 3rd group and last one do not share same opinions with each other but we identified users with same profile characteristics such as authors with same marital status and education.

Clustering Query : "Find groups of users according to username whose opinions are close to those expressed by a set of users, which are related to one or more topics."

![Clustering Query](image)

Figure 6.9: Clustering Query : Sharing similar views with others.

Description of the query execution : In this execution, the author has chosen to find groups of users according to username who share same views with specific other users, related to specific topics as it is shown from the screenshots. If there are not any results, the system return "No debate analysis results were found. Please try again!" for this searching applied.

Analysis Results
CHAPTER 6. A SYSTEM USE CASE

Figure 6.10: Results: Sharing similar views with others.

Description of the opinions analysis results: According to the results, there was found 1 group of users where their opinions are close to the following positions: 

- Studies have shown if you play violent video games you have bad behavior
- Violent TV shows contain many bad scenes which make the children easily commit crimes.

Clustering Query: "Find groups of users with similar opinions with a set of positions who share specific profile characteristics according to one or more topics."

Figure 6.11: Clustering Query: Same users with similar views.

Description of the query execution: In this execution, the author has chosen to find groups of users with similar opinions that have specific profile characteristics, related to topics as it is shown from the screenshots. If there are not any results, the system return "No debate analysis results were found. Please try again!" for this searching applied.
Analysis Results

Description of the opinions analysis results: According to the results, there was found 1 group of users (females) where their opinions are close to the following positions: Studies have shown if you play violent video games you have bad behavior and Violent TV shows contain many bad scenes which make the children easily commit crimes.

Clustering Query: "Find groups of users who agree or disagree on set of positions that have received specific score."

Description of the query execution: In this execution, the author has chosen to find groups of users that express their comments that are associated with specific positions in the form of agreement or disagreement according to the score the
positions have. If there are not any results, the system return "No debate analysis results were found. Please try again!” for this searching applied.

Analysis Results

Figure 6.14: Results : Users with different views.

Description of the opinions analysis results: According to the results, there were found 5 different groups where:

- The 1st and the 5th groups consist of users with similar answers such as support or attack in specific positions but none of these users share similar opinions with each other.

- The 2nd group disagree with the position Studies have shown if you play violent video games you have bad behavior and agree with position Violent video games cause rudeness and many other behavioral problems especially to young people.

- The 3rd group also disagree with the following position Studies have shown if you play violent video games you have bad behavior and agree with the position In my experience video games do not increase violence. They do more good than bad.

- The 4th group only disagree with the following positions In my experience video games do not increase violence. They do more good than bad and Violent video games cause rudeness and many other behavioral problems especially to young people.

Clustering Query : "Find groups of positions which are expressed by users with similar preferences and profile characteristics according to one or more topics.”
6.1. MAIN FEATURES OF APOPSIS PLATFORM

Description of the query execution: In this execution, the author has chosen to find groups of positions which are expressed by users that share same profile characteristics or positions. If there are not any results, the system return "No debate analysis results were found. Please try again!" for this searching applied.

Analysis Results

Description of the opinions analysis results: According to the results, there was found 1 group of users that share same characteristics such as (gender and age) but there were not found similar positions expressed by these users.

Clustering Query: "Find groups of topics according to users agreement/disagreement on positions, which are related with specific issue."
Description of the query execution: In this execution, the author has chosen to find groups of topics according to the users agreements and disagreements when they response to existing comments. If there are not any results, the system return "No debate analysis results were found. Please try again!" for this searching applied.

Analysis Results

- The 1st group do not share common positions and for that reason we do not get common topics from this group. These users are included in the same group because they were selected by the author who made the opinion analysis.
6.1. MAIN FEATURES OF APOPSIS PLATFORM

- The 2nd group share similar opinions and as results we have common topics. We get group of topics that are attacked by other comments and they are *increase violence* and *corrupts teenagers minds*. Other topics that are supported are the following *escaping from reality* and *stress relievers*.

- The 3rd and the 4th group also share similar opinions and so there are groups of topics that are supported by different users. These topics are the following *increase violence* and *corrupts teenagers minds*.

6.1.6 Semantic Querying

The system applies semantic querying for retrieving the supporting or attacking arguments. We implemented six different searching types of querying that allow users to extract different aspects of a dialogue (such as opinions, topics, authors). Next, we provide instances for each one of searching type by showing how the system performs semantic querying.

**Query:** "Search for arguments that can support/attack positions or arguments which are associated with specific aspects".

![Figure 6.19: An instance of searching type 1.](image)

**Description of the query execution:** In this execution, the author has chosen to search for arguments that support any of the selected positions according to the aspect "Insufficient". We expect the results to return arguments in support of preferred positions. Otherwise, the system return "No arguments available" for this searching applied.
Searching Results

Query: "Search for arguments that are supported or attacked by other arguments which are associated with specific aspects.

Description of the query execution: In this execution, the author has chosen to search for arguments that are attacked by other arguments according to the aspect "Insufficient". We expect the results to return arguments that are attacked by other arguments and they can be either positive or negative arguments. Otherwise, the system returns "No arguments available" for this searching applied."
6.1. MAIN FEATURES OF APOPSIS PLATFORM

Searching Results

![Figure 6.22: Results of searching type 2.](image)

**Query**: "Search for arguments or positions that have received specific score.

![Figure 6.23: An instance of searching type 3.](image)

**Description of the query execution**: In this execution, the author has chosen to search for positions that have specific strength value according to what was selected. We expect the results to return positions according to specific scores values. Otherwise, the system return "No arguments available" for this searching applied.
Searching Results

![Figure 6.24: Results of searching type 3.](image1)

**Query:** "Search for arguments or positions that are posted by specific author.

![Figure 6.25: An instance of searching type 4.](image2)

**Description of the query execution:** In this execution, the author has chosen to search for arguments that are posted by specific authors. We expect the results to return arguments according to the authors answers either positive or negative answers. Otherwise, the system return "No arguments available" for this searching applied.
6.1. MAIN FEATURES OF APOPSIS PLATFORM

Searching Results

Figure 6.26: Results of searching type 4.

**Query**: "Search for arguments or positions that have been evaluated as the best arguments according to specific issue.

Figure 6.27: An instance of searching type 5.

**Description of the query execution**: In this execution, the author has chosen to search for positions related to specific issue. We expect the results to return positions which are associated with specific topic of discussion. Otherwise, the system return "No arguments available" for this searching applied.
Searching Results

**Figure 6.28: Results of searching type 5.**

**Query:** "Search for authors whose arguments have received the best score, according to specific issue.

**Figure 6.29: An instance of searching type 6.**

**Description of the query execution:** In this execution, the author has chosen to search for authors whose arguments have received the highest score value. We expect the results to return positions which are associated with specific topic of discussion and their strength is the highest. Otherwise, the system returns "No arguments available" for this searching applied.
Searching Results

Figure 6.30: Results of searching type 6.
Chapter 7

Conclusion and Future Work

As more and more opinions are presented every day on the Web and mostly in discussion forums, blogs and debate portals, the need for analyzing the arguments becomes all the more important in order to understand the validity and the reliability of individual opinions expressed within dialogue. This work addresses this problem by providing a Web-based Platform, called APOPSIS, that motivate users to participate in structured, goal-oriented debates and especially analyze the various aspects of an ongoing dialogue by 1) extracting the different opinions expressed in discussions, 2) determining the most relevant opinions and finally, 3) identifying trends of users that share common characteristics, according to the arguments expressed within debate or user profile characteristics.

We examined the state of the art tools that focus on facilitating dialogues, considering querying, analysis and evaluation of arguments. Each one of these systems encourage participants to be involved in structured debates but none of them provide opinions analysis and help contributors understanding the dynamic flow of debates. Among all systems we distinguished two specific systems that have similar orientation with our implementation, the Queastio-it.com, which provides a method for modeling and analyzing social discussions and e-Dialgos, which enables users to participate and express their opinions in deliberation processes such as city problems, a specific domain where our system, APOPSIS, can be applied effectively. Our work is the first study that is focused not only in structured, goal-oriented debates but mainly provide different means of analysis of users opinions, considering different aspects of dialogues throughout the decision-making process.

Furthermore, we described the main contributions of this work which include; creating new topics of discussion, evaluating the strength value of each argument, analyzing the opinions expressed in a dialogue and finally, semantic querying is applied for retrieving the supporting/attacking arguments. In the first interaction with the system, users may navigate among different existing dialogues or create new debates by initializing a new conversation. Debates in Apopsis, proceed in two different stages, facilitated by a class of users, known as ”moderators”, that ensure the high quality of opinions analysis. In the first stage, users can contribute with
each other by expressing their views in the form of agreement or disagreement, or by voting on arguments where the conversation evolves in a more free way. The system applies a formal framework, called sm-Dice, for evaluating the strength value of each argument and also provide an ontology for describing the conceptual components of debates by storing and retrieving arguments in the form of RDF statements in the Virtuoso repository. In the next stage, the system applies opinions analysis, considering arguments with high strength and user profile characteristics throughout the decision-making process. The platform implemented six different information needs, emerging from users during discussion process. For each one of these information needs, we used a set of ML algorithms which include; the clustering of features (opinions or profile characteristics) and the extraction of association rules, such as K-means, Expectation-Maximization (EM) and the Apriori algorithm. Finally, the system applies semantic querying for retrieving supporting or attacking opinions by applying SPARQL queries.

Regarding future work and research, there are several aspects that are worth investigating. One important aspect is the visualization of the various aspects of opinion analysis by using visualization tools that represent the different information needs, emerging from users during the decision-making process. Moreover, we would like to improve our implementation method in order to have good scalability and performance of our debate analysis. Another idea would be to apply different methods for the parameters used in K-means and Apriori both. Finally, a future direction would be to evaluate our system with real users and provide large datasets of discussions in order to record how efficiently can be applied parameterization algorithms in the system.
Bibliography


Chapter 8

Appendix A

Next, we describe all Argumentation tools and Social Web systems that exist in literature, mostly based on the following papers (Schneider, Jodi and Groza, Tudor and Passant, Alexandre, 2013) [57] and (Scheuer, Oliver and Loll, Frank and Pinkwart, Niels and McLaren, Bruce M, 2010) [55].

<table>
<thead>
<tr>
<th>Tools</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Mapping Schema</th>
<th>Debate</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Questio-it.com [17]</td>
<td>A platform used for offering concrete solutions for both research problems and industrial settings</td>
<td>Extended Social Abstract Argumentation Framework (ESAAF) for evaluating the best answers</td>
<td>OWL ontology</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Graph based</td>
</tr>
<tr>
<td>Debate.org</td>
<td>A social network for creating and participating in debates</td>
<td>Yes</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based discussion</td>
</tr>
<tr>
<td>iDebate</td>
<td>A platform for debating on a topic by providing statements for or against an initial position</td>
<td>set of voting types but not voting mechanism</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based discussion</td>
</tr>
<tr>
<td>CreateDebate</td>
<td>A social debate community for participating in debates</td>
<td>based on user-applied tags</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based discussion</td>
</tr>
</tbody>
</table>

Table .1: Argumentation Tools(1)
<table>
<thead>
<tr>
<th>Tools</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Mapping Schema</th>
<th>Debate</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apopsis</td>
<td>A platform for analyzing opinions</td>
<td>s-mDice,a framework for evaluating arguments</td>
<td>RDF representation, influenced from IBIS model used</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Thread based discussion</td>
</tr>
<tr>
<td>e-Dialgos [1]</td>
<td>A debating platform that motivate users to participate and vote about city planning policies</td>
<td>Yes</td>
<td>no available information</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Thread based discussion</td>
</tr>
<tr>
<td>StackExchange</td>
<td>The purpose is to enable users to post questions and answer them</td>
<td>Viting and Reputation mechanism</td>
<td>no information provided</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Sentence based discussion, questions and answers.</td>
</tr>
<tr>
<td>Quora</td>
<td>An online collaborative information sharing and learning platform</td>
<td>Yes</td>
<td>no information provided</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Thread based conversation.</td>
</tr>
<tr>
<td>DebateGraph</td>
<td>A platform that visualize the debates,collaborate on complex issues</td>
<td>No</td>
<td>IBIS model</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph based, Thread based conversation, private map, navigation representation.</td>
</tr>
<tr>
<td>Debatewise</td>
<td>A web-based platform created with the aim to inform the users who like to make decisions on the basis of gathering information</td>
<td>Yes</td>
<td>no mechanism provided</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Thread based discussion.</td>
</tr>
<tr>
<td>ConvinceMe [50][54]</td>
<td>An online competitive debating environment where users participate in real debates</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Graph based</td>
</tr>
<tr>
<td>DiscourseDB</td>
<td>An online database of opinions and commentary</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Frames based</td>
</tr>
<tr>
<td>LivingVote</td>
<td>A debating platform where users can vote and post arguments</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Pro and Con representation of arguments.</td>
</tr>
</tbody>
</table>

Table 2: Argumentation Tools(2)
<table>
<thead>
<tr>
<th>Tools</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Mapping Schema</th>
<th>Debate</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Araucaria [52]</td>
<td>A software system for analyzing textual arguments</td>
<td>No</td>
<td>Standard, Toulmin and Wigmore schemes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Tree structure and diagram representation.</td>
</tr>
<tr>
<td>Aquanet [39]</td>
<td>A browser-based tool that allows users to graphically represent information for exploring its structure</td>
<td>No</td>
<td>Configurable ontology, composing schemes (NoteCard, gIBIS and Toulmin)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Frames based representation.</td>
</tr>
<tr>
<td>ArguNet [56]</td>
<td>A desktop tool for collaborative argumentation analysis</td>
<td>No</td>
<td>Theory of dialectical structures</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Diagram based representation, argument maps.</td>
</tr>
<tr>
<td>ARGUNAUT [14]</td>
<td>A project of two existing platform, aims at unifying awareness and feedback mechanism in e-discussions environment</td>
<td>No</td>
<td>Pre-defined typology or scheme(e.g critical questions)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Graphical representation.</td>
</tr>
<tr>
<td>AVER [66]</td>
<td>A web-based collaborative system that supports argument construction and argumentation schemes</td>
<td>No</td>
<td>AIF core ontology, argumentation schemes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Multiple views (graphs, linear, table, report view, argument summary).</td>
</tr>
<tr>
<td>AVERs [5]</td>
<td>A sense-making software where human crime investigator can visualise possible stories</td>
<td>No</td>
<td>Anchors narratives theory, argumentation schemes, combination of two kind of reasoning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Graphical, diagram representation.</td>
</tr>
<tr>
<td>ArgVis [23]</td>
<td>An argument visualization tool designed to systematize the argumentation process by representing structured arguments</td>
<td>Yes</td>
<td>SIOC ontology, IBIS model</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph based representation, navigation mechanism.</td>
</tr>
<tr>
<td>Avicenna [47]</td>
<td>A Web-based system for authoring and querying argument structures</td>
<td>No</td>
<td>OWL ontology, Walton scheme</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Arguments based representation.</td>
</tr>
<tr>
<td>ArgDF [70] [49]</td>
<td>A Semantic Web-based platform for representing arguments and building interlinked and constantly evolving argument networks</td>
<td>No</td>
<td>AIF-RDF ontology</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Arguments based representation.</td>
</tr>
<tr>
<td>Tools</td>
<td>Purpose</td>
<td>Voting-Score system</td>
<td>Mapping Schema</td>
<td>Debate</td>
<td>Decision-making process</td>
<td>Opinion Analysis</td>
<td>Visualization</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>---------</td>
<td>-------------------------</td>
<td>------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Carneades [19]</td>
<td>A set of Open Source software tools providing support for a range of argumentation tasks (construction, evaluation and visualization)</td>
<td>No</td>
<td>Critical questions, dialogical aspects of argumentation, persuasion dialogues, mathematical and computational model, Walton's theory, LKIF</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Graphical representation.</td>
</tr>
<tr>
<td>TruthMapping</td>
<td>A Web-based system for capturing conversation in a group</td>
<td>Yes</td>
<td>Beardsley approach</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Sentence based representation.</td>
</tr>
<tr>
<td>Deliberatorium [25]</td>
<td>A conceptual argument mapping tool that organizes discussion</td>
<td>No</td>
<td>IBIS model</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based discussions.</td>
</tr>
<tr>
<td>HERMES [21]</td>
<td>A decision-support system, facilitating communication and recommending solutions</td>
<td>No</td>
<td>IBIS model, extension of Zeno framework</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based discussions.</td>
</tr>
<tr>
<td>IBIS/gIBIS [11]</td>
<td>An application-specific hypertext system designed to facilitate the capture of early design deliberations</td>
<td>No</td>
<td>IBIS model</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph based discussions.</td>
</tr>
<tr>
<td>Questmap [6]</td>
<td>A collaborative tool that is designed to mediate discussions by creating visual information maps</td>
<td>No</td>
<td>IBIS model (many additional nodes types)</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Multiple information maps.</td>
</tr>
<tr>
<td>Rationale [68]</td>
<td>A commercial package allowing the diagramming and visualization of arguments</td>
<td>No</td>
<td>Legal reasoning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Diagram based representation.</td>
</tr>
<tr>
<td>Reason!Able [67]</td>
<td>An educational software that supports argument mapping to teach reasoning skills</td>
<td>Yes</td>
<td>Reasoning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Diagram based, tree representation.</td>
</tr>
<tr>
<td>Cope-it! [22]</td>
<td>A Web-based tool that is designed for community deliberation, and provides a way to synchronise views between IBIS graphs</td>
<td>Yes</td>
<td>Formal argumentation, Reasoning mechanism, IBIS-like formalism</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph or outline representation, navigation mechanism.</td>
</tr>
</tbody>
</table>

Table 4: Argumentation Tools(4)
<table>
<thead>
<tr>
<th>Tools</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Mapping Schema</th>
<th>Debate</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>DiaLaw [32]</td>
<td>An issue-based system that make users understand and constructive logical arguments against other claims</td>
<td>No</td>
<td>Legal justification, dialogue game</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread representation.</td>
</tr>
<tr>
<td>Cohere [59]</td>
<td>A system that provides enough structure to support argument analysis and visualization</td>
<td>No</td>
<td>IBIS, Toulmin, Walton schemes and relax constraints</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph representation of ideas.</td>
</tr>
<tr>
<td>Athena [38]</td>
<td>An argument mapping software, designed to support argument analysis and production of reasoning and argumentation</td>
<td>Yes</td>
<td>Informal logic and reasoning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Tree representation.</td>
</tr>
<tr>
<td>Trellis [18] [31]</td>
<td>A system that was designed to help capture argumentation, grounded in documents</td>
<td>Yes</td>
<td>No information provided</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Tables, Graphs, Tree representation.</td>
</tr>
<tr>
<td>SEAS [37] [36]</td>
<td>A template-based structured argumentation tool originally designed for collaborative intelligence analysis</td>
<td>No</td>
<td>Evidential reasoning, relevance to legal reasoning</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Template based representation, Starburst graphic, Constellation, Tabular and Textual summary.</td>
</tr>
<tr>
<td>TOAST [60]</td>
<td>An implementation of ASPIC+, which allows a structured argumentation system to be processed into arguments</td>
<td>No</td>
<td>ASPIC framework, AIF, Dung framework, different semantics exist</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Textual and visual commentaries.</td>
</tr>
<tr>
<td>Collaboratorium [27]</td>
<td>A prototype Web-based that is enabling collaborative deliberation around the complex systemic challenges, such as climate change</td>
<td>Yes</td>
<td>IBIS model</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Thread based representation.</td>
</tr>
<tr>
<td>Compendium [42] [43]</td>
<td>A knowledge cartography platform, designed to support interoperability with other tools</td>
<td>No</td>
<td>Free-form discussions using the IBIS model</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Tree representation.</td>
</tr>
<tr>
<td>Deme [13]</td>
<td>A platform for online deliberation</td>
<td>No</td>
<td>No information provided</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Document, sentence representation.</td>
</tr>
</tbody>
</table>

Table 5.5: Argumentation Tools(5)
<table>
<thead>
<tr>
<th>Tools</th>
<th>Purpose</th>
<th>Voting-Score system</th>
<th>Mapping Schema</th>
<th>Debate</th>
<th>Decision-making process</th>
<th>Opinion Analysis</th>
<th>Visualization</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEMOS [15]</td>
<td>A system used for “online-democracy” facilitating debates among participants</td>
<td>No</td>
<td>Zeno framework</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Matrix representation.</td>
</tr>
<tr>
<td>SIBYL [30]</td>
<td>A system that supports group decision making by representing and managing the qualitative aspects of decision making processes</td>
<td>No</td>
<td>IBIS model</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Matrix structure.</td>
</tr>
<tr>
<td>Belvedere</td>
<td>A tool that is designed to support scientific argumentation skills in students and to stimulate discussions on scientific topics among students</td>
<td>No</td>
<td>Beardsley style, arguments in term of inference trees</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graphical environment, matrix representation.</td>
</tr>
<tr>
<td>DREW [12]</td>
<td>A collaborative Java environment for teaching argumentation through the Internet</td>
<td>No</td>
<td>Toulmin representation, dialogical reasoning</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph and text editor, chat, ALEX.</td>
</tr>
<tr>
<td>LASAD [33]</td>
<td>A general, cross domain framework that enables users to configure workspaces according to their specific requirements</td>
<td>No</td>
<td>No information provided, applicable in different domains</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph based and table based representation.</td>
</tr>
<tr>
<td>Digalo [58]</td>
<td>A tool for supporting argumentative discussions</td>
<td>No</td>
<td>Logical deductive reasoning, Configurable ontology, a form of rhetorical speech</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Graph based representation, thread discussion.</td>
</tr>
<tr>
<td>Room 5 [34]</td>
<td>A tool that provides a mechanism for studying a broad community’s willingness to perform structured legal argumentation</td>
<td>No</td>
<td>Dialogue game, legal argumentation</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Frame based representation.</td>
</tr>
<tr>
<td>AcademicTalk</td>
<td>An Open Source tool structures and scaffolds the interactions of students during discussion</td>
<td>No</td>
<td>Dialogue game rules</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Thread based representation.</td>
</tr>
</tbody>
</table>

Table 6: Argumentation Tools(6)
Appendix B

1 Argumentation Schema Analysis

1.1 Class and Property Hierarchies

Although they do not provide comprehensive definitions, compact monohierarchi-
cal presentations of the class and property IsA hierarchies have been found to
significantly aid in the comprehension and navigation of the model, and are there-
fore provided below.

The class hierarchy presented below has the following format:

- Each line begins with a unique class identifier, consisting of a number pre-
  ceded by the letter “O”.
- A series of hyphens (“---”) follows the unique class identifier, indicating the
  hierarchical position of the class in the IsA hierarchy.
- The English name of the class appears to the right of the hyphens.
- The index is ordered by hierarchical level, in a “depth first” manner, from
  the smaller to the larger subhierarchies.
- Classes that appear in more than one position in the class hierarchy as a
  result of multiple inheritance are shown in an italic typeface.

The property hierarchy presented below has the following format:

- Each line begins with a unique property identifier, consisting of a number
  preceded by the letter “P”.
- A series of hyphens (“---”) follows the unique property identifier, indicating
  the hierarchical position of the property in the IsA hierarchy.
- The English name of the property appears to the right of the hyphens.
- The domain class for which the property is declared.
- The range class that the property references.
CHAPTER 8. APPENDIX A

- The index is ordered by hierarchical level, in a “depth first” manner, from the smaller to the larger subhierarchies, and by property number between equal siblings.

- Properties that appear in more than one position in the property hierarchy as a result of multiple inheritance are shown in an italic typeface.

1.2 Classes Hierarchy

![Classes Hierarchy](image)

1.3 Class Declaration

The classes are comprehensively declared in this section using the following format:

- Class names are presented as headings in bold face, preceded by the class's unique identifier.

- The line “Subclass of:” declares the superclass of the class from which it inherits properties.

- The line “Superclass of:” is a cross-reference to the subclasses of this class.

- The line “Scope note:” contains the textual definition of the concept the class represents.

- The line “Properties:” declares the list of the class’s properties.
Each property is represented by its unique identifier, its forward name, and the range class that it links to, separated by colons.

Inherited properties are not represented.

**O1 Creation Data**
- Subclass of: -
- Superclass of: **04 Issue**
- **05 Topic**
- **06 Argument**
- Properties: 
  - P3 datetime: rdfs:Resource
  - P9 created_by: schema:Person
- Scope note: This class comprises information about the provenance of the object such as the datetime along with the person which created it.

Figure 2: Creation Data.

**O2 Strength**
- Subclass of: -
- Superclass of: **05 Topic**
- **06 Argument**
- Properties: 
  - P4 voting: rdfs:Resource
  - P5 dialogue: rdfs:Resource
  - P6 congruence: rdfs:Resource
  - P7 acceptance: rdfs:Resource
  - P8 quality: rdfs:Resource
- Scope note: This class comprises information required for the score calculation of arguments i.e., comments, positions etc.

Figure 3: Strength.

**O3 Content**
- Subclass of: -
- Superclass of: **04 Issue**
- **05 Topic**
- **06 Argument**
- Properties: 
  - P1 has_content_value: rdfs:Resource
  - P2 has_content_type: rdfs:Resource
- Scope note: This class comprises information about the content contained by each object. Each content object has a type and a value

Figure 4: Content.
**O4 Issue**
Subclass of: 01 Creation_Data  
03 Content  
Superclass of: -  
Properties: P10 consists_of: 05 Topic  
P21 category 013 Category  
P23 has_reviews rdfs:Resource  
P27 closed_time rdfs:Resource  
Scope note: This class is the main part of debate and it can be either a question or a statement that need to be answered by participants in the discussion. An issue consists of one or more topics.

Figure .5: Issue.

**O5 Topic**
Subclass of: 01 Creation_Data  
02 Strength  
03 Content  
Superclass of: -  
Properties: P10 is_characteristic_of: rdfs:Resource  
Scope note: A topic is specific tag which is closely related to an issue. Contributors can agree or disagree on specific tag of an issue.

Figure .6: Topic.

**O7 Position**
Subclass of: 06 Argument  
Superclass of: -  
Properties: P20 status rdfs:Resource  
P22 related_To 04 Issue  
P26 belongs_To rdfs:Resource  
Scope note: This class comprise the direct opinions about topics on specific issue. A position can be considered as an argument that can not support or attack another position.

Figure .7: Position.
1. ARGUMENTATION SCHEMA ANALYSIS

**O6 Argument**
- Subclass of: O1 Creation_Data
  - O2 Strength
  - O3 Content
- Superclass of: O7 Position
  - O8 Comment
- Properties:
  - RP11 about_topic: O5 Topic
  - P12 attacks: O6 Argument
  - P13 supports: O6 Argument
  - P14 replies: O6 Argument
  - P17 contains: O9 Rating
  - P30 has_conclusion: O13 Proposition
- Scope note: This class represents a set of statements created by users which comprise their opinion about one or more topics. Arguments can also be considered as answers which e.g., attack, supports etc. other arguments and so on.

Figure .8: Argument.

**O8 Comment**
- Subclass of: O6 Argument
- Superclass of: -
- Properties:
  - RP15 related_to_position: O7 Position
  - P20 status: rdfs:Resource
- Scope note: This class is considered an extension of class Argument which can be used to represent the dialogues created under positions for certain topics.

Figure .9: Comment.

**O9 Rating**
- Subclass of: -
- Superclass of: O10 Vote
  - O11 Star-Based
- Properties:
  - P16 rates: O5 Topic
  - P19 has_rating: rdfs:Resource
- Scope note: This class comprises information for a general rating system which can be used upon topics, comments etc. The rating system may have various forms such as 5-stars system, positive/negative votes etc.

Figure .10: Rating.
### O10 Vote

<table>
<thead>
<tr>
<th>Subclass of:</th>
<th>09 Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Superclass of:</td>
<td>-</td>
</tr>
<tr>
<td>Properties:</td>
<td>-</td>
</tr>
<tr>
<td>Scope note:</td>
<td>This class represents the special case of a rating system which refers on votes. The votes can be either positive, or negative.</td>
</tr>
</tbody>
</table>

Figure .11: Vote.

### O11 Star-Based

<table>
<thead>
<tr>
<th>Subclass of:</th>
<th>09 Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Superclass of:</td>
<td>-</td>
</tr>
<tr>
<td>Properties:</td>
<td>-</td>
</tr>
<tr>
<td>Scope note:</td>
<td>This class represents the special case of a rating system which refers on stars. The maximum number of stars is application specific and highly customizable.</td>
</tr>
</tbody>
</table>

Figure .12: Star Based.

### O12 Category

<table>
<thead>
<tr>
<th>Subclass of:</th>
<th>01 Creation_Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Superclass of:</td>
<td>-</td>
</tr>
</tbody>
</table>
| Properties: | P1 has_content_value rdfs:Resource  
P2 has_content_type rdfs:Resource  
P3 datetime rdfs:Resource  
P9 created_by schema:Person |
| Scope note: | This class groups every debate/issues into categories. Every issue according to the subject of discussion is included by the user of the system into specific category. |

Figure .13: Category.
1. ARGUMENTATION SCHEMA ANALYSIS

013 Proposition

Subclass of: 01 Creation_Data
03 Content

Superclass of: -

Properties:
P1 has_content_value rdfs:Resource
P2 has_content_type rdfs:Resource
P3 datetime rdfs:Resource
P4 created_by schema:Person
P28 in_conflict: 013 Proposition
P29 premise_of: 06 Argument

Scope note: This class represents the minimum unit, from which an argument is composed. It attaches the structure of <premises, conclusion> to an argument, with the parts of premises, conclusion to be Propositions.

Figure .14: Proposition.

1.4 Properties Hierarchy

<table>
<thead>
<tr>
<th>Property id</th>
<th>Property Name</th>
<th>Entity – Domain</th>
<th>Entity – Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>has_content_value</td>
<td>03 Content</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P2</td>
<td>has_content_type</td>
<td>03 Content</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P3</td>
<td>datetime</td>
<td>01 Creation_Data</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P4</td>
<td>voting</td>
<td>02 Strength</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P5</td>
<td>dialogue</td>
<td>02 Strength</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P6</td>
<td>congruence</td>
<td>02 Strength</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P7</td>
<td>acceptance</td>
<td>02 Strength</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P8</td>
<td>quality</td>
<td>02 Strength</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P9</td>
<td>created_by</td>
<td>01 Creation_Data</td>
<td>schema:Person</td>
</tr>
<tr>
<td>P20</td>
<td>consists_of</td>
<td>06 Issue</td>
<td>05 Topic</td>
</tr>
<tr>
<td>R011</td>
<td>about_topic</td>
<td>06 Argument</td>
<td>05 Topic</td>
</tr>
<tr>
<td>R012</td>
<td>attacks</td>
<td>05 Argument</td>
<td>06 Argument</td>
</tr>
<tr>
<td>P13</td>
<td>supports</td>
<td>06 Argument</td>
<td>06 Argument</td>
</tr>
<tr>
<td>P14</td>
<td>replies</td>
<td>06 Argument</td>
<td>06 Argument</td>
</tr>
<tr>
<td>R015</td>
<td>related_to_position</td>
<td>07 Position</td>
<td>05 Position</td>
</tr>
<tr>
<td>P16</td>
<td>rates</td>
<td>09 Rating</td>
<td>05 Topic</td>
</tr>
<tr>
<td>P17</td>
<td>contains</td>
<td>06 Argument</td>
<td>09 Rating</td>
</tr>
<tr>
<td>P18</td>
<td>is_characteristic_of</td>
<td>05 Topic</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P19</td>
<td>has_rating</td>
<td>09 Rating</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P20</td>
<td>status</td>
<td>08 Comment</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P21</td>
<td>category</td>
<td>04 Issue</td>
<td>012 Category</td>
</tr>
<tr>
<td>R022</td>
<td>related_to</td>
<td>07 Position</td>
<td>04 Issue</td>
</tr>
<tr>
<td>P23</td>
<td>has_reviews</td>
<td>04 Issue</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P24</td>
<td>has_education</td>
<td>schema:Person</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P25</td>
<td>marital_status</td>
<td>schema:Person</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P26</td>
<td>belongs_to</td>
<td>07 Position</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P27</td>
<td>closed_time</td>
<td>04 Issue</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>P28</td>
<td>in_conflict</td>
<td>013 Proposition</td>
<td>013 Proposition</td>
</tr>
<tr>
<td>P29</td>
<td>premise_of</td>
<td>013 Proposition</td>
<td>06 Argument</td>
</tr>
<tr>
<td>P30</td>
<td>has_conclusion</td>
<td>05 Argument</td>
<td>013 Proposition</td>
</tr>
</tbody>
</table>

Figure .15: Properties Hierarchy.
1.5 Property Declaration

The properties are comprehensively declared in this section using the following format:

- Property names are presented as headings in bold face, preceded by unique property identifiers.
- The line “Domain:” declares the class for which the property is defined.
- The line “Range:” declares the class to which the property points, or that provides the values for the property.
- The line “Superproperty of:” is a cross-reference to any subproperties the property may have.
- The line “Scope note:” contains the textual definition of the concept the property represents.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1 has_content_value</td>
<td>This property identifies the value of the content an argument can have. In most cases the value is plain text, but it can also be a video, an image, a music track etc.</td>
</tr>
<tr>
<td>P2 has_content_type</td>
<td>This property identifies the corresponding type of the content. For the content type we have an extensible set of values such as: TEXT, IMAGE, VIDEO, EMPTY</td>
</tr>
</tbody>
</table>

Figure .16: has content value property.

Figure .17: has content type property.
### P3 datetime

<table>
<thead>
<tr>
<th>Domain:</th>
<th>O1 Creation_Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range:</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>Scope note:</td>
<td>This property identifies the datetime an object (e.g., an argument, a topic, etc.) was created.</td>
</tr>
</tbody>
</table>

Figure .18: datetime property.

---

### P4 voting

<table>
<thead>
<tr>
<th>Domain:</th>
<th>O2 Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range:</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>Scope note:</td>
<td>This property identifies the value of the voting strength of an argument or topic. This value is used for the calculation of the acceptance and quality of the argument.</td>
</tr>
</tbody>
</table>

Figure .19: voting property.

---

### P5 dialogue

<table>
<thead>
<tr>
<th>Domain:</th>
<th>O2 Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range:</td>
<td>rdfs:Resource</td>
</tr>
<tr>
<td>Scope note:</td>
<td>This property identifies the value of the dialogue strength of an argument or topic. This value is used for the calculation of the acceptance and quality of the argument.</td>
</tr>
</tbody>
</table>

Figure .20: dialogue property.
P6 congruence

Domain:  
Range:  
Scope note: This property identifies the value of the congruence strength of an argument or topic. This value is used for the calculation of the acceptance and quality of the argument.

Figure .21: congruence property.

P7 acceptance

Domain:  
Range:  
Scope note: This property identifies the value of the acceptance score of an argument or topic.

Figure .22: acceptance property.

P8 quality

Domain:  
Range:  
Scope note: This property identifies the value of the quality score of an argument or topic.

Figure .23: quality property.
1. ARGUMENTATION SCHEMA ANALYSIS

**P9 created_by**

Domain: 01 Creation_Data  
Range: schema:Person  
Scope note: This property identifies the user who created a specific argument (i.e., comment/position) for one or more topics.

Figure .24: created by property.

**P10 consists_of**

Domain: 04 Issue  
Range: 05 Topic  
Scope note: This property connects an issue of discussion with one or more topics which refer to the said issue.

Figure .25: consists of property.

**RP11 about_topic**

Domain: 06 Argument  
Range: 05 Topic  
Scope note: This property connects any argument with one or more topics of discussion. This redundant property is useful in cases where we have dialogues of comments about topics as it provides a direct association between the comments and the discussed topics. This leads to a more efficient reasoning over the ontology.

Figure .26: about topic property.
**P12 attacks**

**Domain:** 06 Argument  
**Range:** 06 Argument  
**Scope note:** This property identifies an attacking relation between two arguments denoting that the subject argument of the property attacks/disagrees with the object.

Figure .27: attack property.

**P13 supports**

**Domain:** 06 Argument  
**Range:** 06 Argument  
**Scope note:** This property identifies a supporting relation between two arguments denoting that the subject argument of the property supports/agrees with the object.

Figure .28: supports property.

**P14 replies**

**Domain:** 06 Argument  
**Range:** 06 Argument  
**Scope note:** This property identifies a simple relation between two arguments denoting that the subject argument of the property replies on the object without denoting if it agrees or disagrees.

Figure .29: replies property.
1. ARGUMENTATION SCHEMA ANALYSIS

**RP15 related_to_position**
- Domain: 08 Comment
- Range: 07 Position
- Scope note: This property connects one or more comments with a position. This is also a redundant property which is useful in cases where we have dialogues of comments which discuss and argue on a specific position. Having a direct comment-to-topic association leads to a more efficient reasoning over the ontology.

Figure .30: related to position property.

**P16 rates**
- Domain: 09 Rating
- Range: 05 Topic
- Scope note: This property identifies a rating relation between a rating object (e.g., vote, star-based) and a topic which is rated.

Figure .31: rates property.

**P17 contains**
- Domain: 06 Argument
- Range: 09 Rating
- Scope note: Taking into account that in many cases, a comment, or a position also rate on or more topics of discussion, this property describes this association between an argument and one or more ratings.

Figure .32: contains property.
### P18 is_characteristic_of

**Domain:** 05 Topic  
**Range:** rdfs:Resource  
**Scope note:** This property identifies the object which is characterized by a specific topic. For example, consider a topic which refers on a facility of a hotel i.e., swimming pool. Using this property, we can denote that this topic is a characteristic of the webpage URL which contains information hotel’s swimming pool.

Figure .33: is characteristics of property.

### P19 has_rating

**Domain:** 09 Rating  
**Range:** rdfs:Resource  
**Scope note:** This property identifies the rating value w.r.t. the rating type which is used. For example, in the case of a voting system, the rating value is +1 for positive votes and -1 for negative. On the other hand, in case of star based systems, the rating value is the number of stars.

Figure .34: has rating property.

### P20 status

**Domain:** 08 Comment  
07 Position  
**Range:** rdfs:Resource  
**Scope note:** This property identifies the status of a comment or position. For the status of the comment we have an extensible set of values such as:
- ACTIVE
- OUTDATED
- DELETED

Figure .35: status property.
1. ARGUMENTATION SCHEMA ANALYSIS

### P21 category

**Domain:** 013 Category  
**Range:** rdfs:Resource  
**Scope note:** This property identifies the different categories where an issue can be included. The web-system contains the following categories:

- Politics
- Law
- Science
- Technology
- Business
- Religion
- Health
- Sports

---

Figure .36: category property.

### P22 related_To

**Domain:** 07 Position  
**Range:** 04 Issue  
**Scope note:** This property describes the association of any position with an Issue according to specific topics. An issue can have one or more positions.

---

Figure .37: related To property.

### P23 has_reviews

**Domain:** 04 Issue  
**Range:** rdfs:Resource  
**Scope note:** This property describes the views that each issue has received according to how many times an author visit or not a debate conversation. There is no restriction of visiting a debate. E.g you do not have to log in first in order to view a conversation.

---

Figure .38: has reviews property.
### P24 has_education

**Domain:** schema:Person  
**Range:** rdfs:Resource  
**Scope note:** This property is associated with the Person class and describes the education level of every registered author within the web-system. The education level consists of three values:
- Primary School
- High School
- University

Figure .39: has education property.

### P25 marital_status

**Domain:** schema:Person  
**Range:** rdfs:Resource  
**Scope note:** This property is associated with the Person class and describes the marital status of every registered author within the web-system. The marital status consists of three values:
- Single
- Married
- Divorced

Figure .40: marital status property.

### P26 belongs_To

**Domain:** 07 Position  
**Range:** rdfs:Resource  
**Scope note:** This property is associated with the Position class and describes the stages of every position. The web-system consists of two stages:
- **Stage1**: Free conversation. Every one who is registered can post an argument
- **Stage2**: Arguments are transported to the next stage according to authors opinions.

Figure .41: belongs To property.
1. ARGUMENTATION SCHEMA ANALYSIS

P27 closed-time

Domain: 04 Issue
Range: rdfs:Resource
Scope note: This property is associated with the Issue class and describes the datetime of a closed debate. A moderator has the advantage of closing a debate when he believes that there is no need any more of keeping a debate active for users.

Figure .42: closed time property.

P28 in conflict

Domain: 013 Proposition
Range: 013 Proposition
Scope note: This property is associated with the Proposition class and describes a conflict between two propositions. In our model, a conflict lies between two mutually confuted contents. For example, two conflicting propositions are the following, "Car X is the best choice" and "Car X is not the best choice".

Figure .43: in conflict property.

P29 premise_of

Domain: 013 Proposition
Range: 06 Argument
Scope note: This property is associated with the Proposition class and describes that a proposition belongs to the premise set of an argument. An argument may have zero or more propositions in its premise set.

Figure .44: premise of property.
### P30 has_conclusion

<table>
<thead>
<tr>
<th>Domain:</th>
<th>06 Argument</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range:</td>
<td>013 Proposition</td>
</tr>
</tbody>
</table>

**Scope note:** This property is associated with the Argument class and describes the relation between an argument and its justified conclusion. Each argument has exactly one conclusion.

Figure .45: has conclusion property.