Thesis Proposal
Using Argument Analysis to Define a
Structure for User Generated Content

Clare Llewellyn

Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2012
Abstract

Conversation is how people compare ideas, solve problems and identify disagreements as part of daily life. The internet contains several good information sources which can provide facts and general information, but it is less obvious where to find meaningful discussions and arguments that are grounded with evidence.

User generated content can be found in social networking sites, add comments sections of news articles, blogs or micro blogs and information collation sites such as Wikipedia. Any evaluations that occur in these discussions must be conducted by humans because any argument structure is not presented explicitly and therefore cannot be automatically analysed. If a user wishes to hear all sides of an argument it can be difficult to find all the right information. This work aims to automatically identify and extract the structure of arguments from user generated content. Thereby, allowing the users of this content to see the structured arguments contained within, with the claims, counter claims, and grounds. Presenting this structure can allow users to more easily navigate the content thereby enabling the user to become part of the conversation in a way that is arguably more useful to them.

The steps in the process, and therefore the problems that this work aims to tackle, are the identification of discussions on different topics, the identification of spans of text that illustrate the discussion, classification of text into an argument structure so as to define the relationships between spans of text and to present this information to users.
Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Clare Llewellyn)
Table of Contents

1 Introduction 1
  1.1 The Problem ........................................... 1
  1.2 Motivation ............................................. 2
  1.3 Objective .............................................. 4

2 Background 7
  2.1 Argument Theory ....................................... 7
  2.2 Argument Data Structures ............................... 10
  2.3 Applications ........................................... 11
  2.4 Techniques for Identifying and Classifying Argumentation in Text . 13
    2.4.1 Classifying Arguments ............................... 13
    2.4.2 Identifying Arguments ............................... 18

3 Approach 27
  3.1 Work In Progress ..................................... 28
    3.1.1 Data Sets .......................................... 28
    3.1.2 Topic Detection ................................... 30
    3.1.3 Argument Analysis and Classification ............... 35
    3.1.4 Conclusions from work so far ........................ 42

4 Future Work 45
  4.1 Approach ............................................... 45
  4.2 Plan .................................................. 46

Bibliography 51
Chapter 1

Introduction

1.1 The Problem

The scientific method requires that empirical and measurable evidence can be found to support a hypothetical explanation of a phenomenon. The hypothesis can be used to predict the outcomes in a certain scenario, which can then be evaluated through evidence. Academics can test their theories through discussion, allowing the evaluation of a hypothesis with regard to evidence provided by others. Conversation, or discourse, is how people compare ideas, solve problems and identify disagreements as part of daily life.

In traditional print media journalists, academics and other writers perform the tasks of collating and curating information to present in a structured way to consumers. These professionals summarise the debates and add in their own claims and evidence. The consumer of this content is then free to consult this evidence and determine their own beliefs. In the Internet domain there is an overwhelming amount of information available, some is collated and curated in this manner but much is not. There are several good information sources which can provide facts and general information but it is less obvious where to find meaningful discussions and arguments that are grounded with evidence.

Tools are freely available that allow users to be both producers and consumers of content and to be part of the wider internet conversation. Users add data to social networking sites, they use the add comments sections of news articles, they blog or microblog and they contribute to articles for information collation sites such as Wikipedia. Any evaluations that occur in these discussions must be conducted by humans because any argument structure is not presented explicitly and therefore cannot be automati-
cally analysed. If a user wishes to hear all sides of an argument (the full argument space) it can be difficult to find all the right information.

1.2 Motivation

The models of presentation in current user generated content systems often mean the context of a post is lost. Looking at information without this context can give a skewed version of the argument. Most user generated content systems allow users to comment in a linear way, presenting comments as a list with each addition being added to the top or the bottom of the list. Additionally many sites allow users to comment upon the comments of other users, thereby creating a thread like structure. Often this data is presented by time of addition, so the user sees the most recently added data first, or by a user rating system, often represented using votes for or against particular posts (comment systems) or by the reposting of information (blogs, Twitter). Unfortunately the speed with which comments are added is extremely variable and this can make it difficult to follow the conversations and for users to comment at the most appropriate locations. If a rating system is used, a post that references or quotes another may appear out of context and the point of the post may be obscured.

Users are generating a large volume of data. The volume of data can be overwhelming therefore users may adopt the approach of only reading what they are interested in. This can lead to cyberpolarization where users with opposing opinions are not aware of the evidence from the opposing side (Faridani et al., 2010). Often the users only look at the most popular or the most recent content, depending on the display system used, and therefore can form a misleading impression of the overall discussion. The current thread style is no longer a good interface for interacting with this user generated data as it can lose the context and variety of the discussion.

It is a generally held belief that users who post moderate, well-thought-out opinions can often be shouted down by extremists and discussions can thus end with neither side appreciating the other’s point of view and even becoming insulting. This can lead to those with interesting opinions not posting as they fear they will either not be read or they will be shouted down and insulted (Faridani et al., 2010).

It can be difficult for a user to determine the most useful location to add their own information so that it can be found by others. Allowing the users of content to see the structure of arguments, with the claims, counter claims, replies and grounds, would allow the users to navigate a large volume of content whilst retaining the context of
the original post. It would enable users to see the full discussion space whilst allowing them to easily disregard comments that are not well thought out or are insulting. This type of structure could facilitate knowledge acquisition and enable the user to become part of the conversation in a way that is arguably more useful to them.

There is no easy way to gain an overview of the opposing sides of a discussion. For example within academia a user must read many journal papers or blogs and have conversations with specialists in the areas they wish to investigate. Conversations can be conducted online through commenting on specific papers or blogs or identifying the experts through social networking tools and contacting them directly. A tool that analyses these conversations to determine the structure of the discussion would make the content easier to navigate and make these conversations more accessible, increasing the usefulness of this content and aiding scholarship.

In essence, argument is a conversation which involves different points of view. The initial point of view is a claim and this claim can be countered with another claim. The claims are backed up with evidence which can also be referred to as grounds. An individual constructs an argument claim that represents their position, and in this they explain to others the evidence that has led to their position (the claim and the grounds that they use to support this). A counter claim challenges the original position, providing further evidence or information. This is intended to cause the composer of the original claim to re-evaluate their position and accept the counter claim. Those who have been involved in the argument have been made aware of other perspectives on the original problem thus allowing the participants to gain new knowledge that may change or confirm their perspective, encourage them to conduct further investigation, or they may use this evidence to solve other problems at a later date.

It is proposed here that user generated discussions can be expressed as arguments with different points defined as claims, counter claims and evidence and that this structure can be automatically extracted from the content and presented to the user for further interaction. Presently, as discussed further in chapter 2, there are techniques that can be used to identify possible argument components in text, and there are tools that can be used to classify these components and standardised structures for storing and displaying argumentation. However these processes have not been integrated and used to solve the problems defined above that relate specifically to user generated content.
1.3 Objective

The objective of this work is to automatically identify and extract the structure of arguments from user generated content, to display this structure and to help a user identify where they should add their own content.

In Chapter 2 there is a discussion on current approaches for extracting arguments from user generated content. This work differs from other areas of argument analysis as it aims to extract arguments from within a stream of conversations from user generated content sources such as Twitter, comments on news articles and other media as yet undefined; therefore specific conversations must be identified and appropriate sections of text extracted. In addition, there is an intention to combine the extraction of the argument structure with tools available for visualisation of these arguments. These tools will be extended to enable the users to add their own content at appropriate locations.

The proposed steps in this process are inspired by fields such as opinion mining. Bal and Saint-Dizier (2009) analyse product reviews to determine current public opinion on commercial products; this work is described in more detail in Chapter 2. The plan for the extraction of the data is adapted from the steps they define. The steps in the process, and therefore the problems that this work aims to tackle, are:

1. Identify discussions on different topics
2. Identify spans of text that illustrate the discussion
3. Classify into a structure so as to define the relationships between spans of text
4. Present this information to users

The objective of this document is to briefly cover the current research that is relevant, to describe the experimentation that has taken place up to this point and to outline and provide a plan for the future direction of this work.

Chapter 2 presents a brief overview of the field of argument theory. The current standards for argument structure and ontologies are surveyed and the applications which are based upon those structures are described. A review of the current literature that describes processing methods for extracting arguments from text has been conducted and is presented in section. This section describes both current machine learning techniques for classifying text into an argument structure and other approaches such as rhetorical structure theory and the theory of argument discourse markers that may provide techniques that will be applicable to this process.

Chapter 3 describes the approach that is being taken to this work. The data set that is currently being used and the work that has been completed is described.
Chapter 4 describes the work that will be conducted in the next two years and a plan is presented for the successful completion of this project.
Chapter 2

Background

The study of argumentation builds upon ideas from many disciplines including logic, dialogue theory, artificial intelligence, agent systems, linguistics, and argument theory. It has application in computer supported collaborative learning, the semantic web and artificial intelligence. In investigating argument analysis, ideas from argument theory, rhetorical structure theory and computer supported collaborative work are considered.

2.1 Argument Theory

Argument theory is the evaluation of conclusions by logically reasoning over claims that are based on premises. In 2008 Douglas Walton defined three major types of argument (Walton et al., 2008), (Rahwan, 2008b):

- Deductive, if the premise is true the conclusion is true
- Inductive, argument supported by generalisations from empirical evidence
- Defeasible (or presumptive), the conclusions are plausible given the premise, but can be proved false by new evidence.

Historically the deductive model was used to evaluate an argument; if the premise can be evaluated as true then the conclusion is therefore true. More recently there has been a shift to a more ‘human’ type of evaluation of arguments where a degree of uncertainty in the conclusion is possible. This is because using deductive logic to evaluate arguments does not work well in certain situations. For example considering the premise that ‘books are normally longer than 1 page’, deductive reasoning could not evaluate this premise as true until all the books that will ever exist have been evaluated in order to determine if they are longer than a page. At this current time
defeasible argumentation methods are preferred when evaluating arguments. Defeasible
argumentation occurs when the conclusion is accepted in the light of present evidence
but can be changed if new evidence is provided, so that ‘books are normally longer
than one page’ can be evaluated as true until many books with a single page are found.
A premise can be assumed to be true until there is something which disproves it. This
allows reasoning in the absence of complete knowledge.

In 1958 Stephen Toulmin noted that natural human argumentation differs from the
logics in deductive arguments but shares the same basis of claims and evidence, and
from this he concluded that formal logic was different from that used in academic dis-
cussion (Toulmin, 2003). He produced a model of this academic style of argumentation
and founded the modern study of argumentation to reflect this. The Toulmin model of
argumentation is applied to practical problems where reasoning is a process of testing
ideas. A claim is provided then justified, for an argument to succeed a good justifica-
tion of this claim must be made. This argumentation model is not as rigorous as those
that use logic but provides enough evidence to allow rational acceptance (Walton et al.,
2008). It is proposed here that this more natural description of argumentation can be
identified in text and used to determine a structure for that content to make it easy to
navigate the conversations in user generated content.

The Toulmin model (Toulmin, 2003) of argumentation can be used to provide a
structure for argument. This flexible approach is based upon human argumentation
which reflects the nature of discussions found in user generated text. The scheme is
made up from 6 components:

1. Claim - A conclusion to be evaluated
2. Ground - Evidence and data, used to support the claim
3. Warrant - A statement authorising the ground from the claim (to bridge the gap
   between claim and ground)
4. Backing - Credentials that certify the statement expressed in the warrant (when
   the warrant itself is not enough / credible)
5. Rebuttal - Recognising the restrictions which may legitimately be applied to the
   claim
6. Qualifier - Expressing the degree of force or certainty of the claim

The aim of analysing text would be to identify the claims provided and to identify
where those claims are justified. For an argument to succeed it should provide good
justification of a claim. For this a claim, ground and warrant are always needed but a
qualifier, backing and rebuttal are not essential. The process would analyse the text to initially identify the claim and ground and when these are found, attempt to find the other components listed above.

Argument structure takes many forms from single premise arguments to linked, convergent, serial and divergent arguments. In a linked argument there are two or more premises that are dependent on each other and when combined they support the argument conclusion. In convergent arguments each premise individually supports the argument. With serial arguments, the conclusion of an argument acts as the premise for another. A divergent argument means that a premise can support multiple argument conclusions. These structures can also be combined to model complex argumentation (Rahwan, 2008b).

Argument theory can be used to automate reasoning, the evaluation of arguments to determine what is true. Schemes are used to present the argumentation structure which can then be automatically evaluated. Walton has spent many years investigating and defining argumentation through a number of these schemes. He provides extensive descriptions of 96 argument schemes that are designed to represent patterns of human argument and reasoning (Walton et al., 2008). Each scheme has a set of critical questions that can be used to reason over the arguments to evaluate it and establish its strength.

Walton also presents dialogue types to provide context for argument schemes. Each type has specific speech events. There are 6 types (Walton, 2000):

1. Persuasion - resolution of a conflict of opinion to resolve or clarify an issue
2. Negotiation - to make a deal that is satisfactory to all participants
3. Inquiry - to find or verify evidence in order to evaluate a hypothesis
4. Deliberation - to decide the best course of action in a practical situation / choice
5. Information seeking - to acquire or give information
6. Eristic - to fight and quarrel without any reasonable goal

Obviously there are criticisms of these approaches. Many real arguments are composites of 2 or more of the dialogue types and of the various structures defined above (Rahwan, 2008b). A way to evaluate an argument, as proposed by Walton et al. (2008), is through the answering of critical questions. As these questions are dependent on each of the argument schemes, there are therefore many. This makes it hard to evaluate using this scheme (van Eemeren et al., 2008, 2012). Critics of Toulmins model highlight that it does not allow methods for reasoning over the argument in order to
evaluate it through the proposing of critical questions, as in Walton's schemes for argumentation (Cartwright and Atkinson, 2009).

In summary it is very difficult to reason and form conclusions automatically. Humans are much better at this type of task than machines. The most efficient approach to this problem may be a system that utilises machines for tasks such as the extraction and summary of the argumentation points and allow humans to evaluate these arguments.

### 2.2 Argument Data Structures

The aim of structuring data is to provide a basic overview that allows users to more easily understand it. When this content contains a discussion or argument the best way to structure the data is to map it to an argument structure. This allows the user to see the main points of the discussion. The claims and the evidence associated with those claims give the user an indication of where the argument is strong and where it is weak. The points where the claims have weak evidence suggest the most likely places for the evolution of the argument.

Much of the work in argumentation structure comes from the education domain where maps are made by students to define arguments on certain topics. These students discuss a given question and provide claims and counter claims and shape this information into a giant map which allows others to visualise the discussion. When this is done manually it can take a very long time to cover all views on a topic. For example, it is estimated that the production of an argument map describing the debate on whether computers think took 7,000 hours to complete (Metzinger, 1999).

Semantic web technologies such as RDF and ontologies are currently used to define the basis for many existing argument systems. The currently available ontologies allow a feature rich representation. The expression of an argument in an ontological structure allows discourse structure and component relationships to be accessible to computation so that the data can be navigated and compared automatically.

The data can be represented in XML, for example, the Argument Markup Language (AML), which is an XML based language for annotating claims and premises (Reed and Rowe, 2004; Rahwan, 2008a). Various semantic web models for argumentation are also available including: a hypertext based approach, the Issue-Based Information Systems (IBIS) (http://hyperdata.org/xmlns/ibis/, 2012), a biological approach the SWAN/SIOC, which is an alignment of two ontologies the Semantic Web Applications in Neuromedicine (SWAN) and Semantically-Interlinked Online Communities
2.3 Applications

(SIOC) (Schneider et al., 2010; Schneider and Passant, 2012) and the ScholOnto ontology (Buckingham Shum et al., 2000) which is used to support debate in digital libraries.

The Argument Interchange Format (AIF) is an interchange format has been proposed for these different ontologies (Rahwan, 2008a). AIF is an ontology of argument related concepts from several different schemes. It is expressed in the Web Ontology Language (OWL) which offers a rich feature set and allows inference using descriptive logics (Mcguinness and van Harmelen, 2004). Currently it only represents a limited set of argument concepts and must be extended to capture other argument formalisations and schemes. In this format an argument is made up of argument entities represented as nodes. Each node has attributes such as title, creator, creation date, certainty, acceptability. It is made up of I-nodes (information) and S-nodes (schemes). The I-nodes represent passive information such as claim, premise, grounds and the S-nodes are the patterns of reasoning. The schemes belong to a class and are classified into types (such as rule of inference scheme, conflict scheme and preference scheme). AIF can also be expressed using the Resource Description Framework (RDF) (Lassila et al., 1998) as AIF-RDF which although less expressive allows interaction with other RDF based technologies (Rahwan, 2008a).

Although many of these formats share a similar underlying RDF structure they are not compatible with each other. There are some indications that AIF-RDF standard may emerge as the standard that is adopted in this domain, since it allows the representation of the largest number of argument schemes. But, the extensive functionality of AIF-RDF also means that users must identify the logical aspects of argumentation, and as this can be very complex, this may limit the uptake of the system. As there is no single unified ontology as yet it is difficult to commit to using a single ontological structure. The adoption of a standard would promote interoperability, but at this point there is a reluctance to commit to one for this work, but rather to remain flexible and adapt to the ontologies as needed for display purposes.

2.3 Applications

Argument structure within content can be defined either manually or automatically. Internet tools have been developed that encourage users to take part in debates (Cartwright and Atkinson, 2009). Generally these tools required the user to add the structure as they add the content. Once added the content is held in a repository and visualisation
techniques are used to express the structure. There are several examples of argument repositories; in general they do not interoperate with each other as they do not use the same underlying format, and the argument structures provided are either very simple or very complex depending upon the purpose they are designed for (Rahwan, 2008a). A shared ontology or format would be required to support integration of arguments from different tools, or an ontology mapping tool would be necessary to align these different formats.

A brief review of some of the tools available are presented below.

1. Debateabase is from the International Debate Education Association, a not-for-profit organisation which promotes, amongst other things, critical thinking. It is a database of argument questions where the pros and cons of the arguments are listed. These pros and cons for each question are written by experts for those new to the debate to learn the arguments. There are for and against points for each debate and counter points to those points. The site is well presented, has a nice look and feel and good functionality, however the arguments themselves are confusing and the logic of the debate is not very clear (www.idebate.org/debatabase).

2. Truthmapping is a volunteer site funded by donations. Users create arguments by stating premises and claims, critiques and rebuttals can be added, the arguments can be chained together and linked to evidence added. The site is used to learn and practice the logic of debates. Therefore in the logic of the site, what is a premise, a claim and conclusion is very clear. The maps of the argument are good. (www.truthmapping.com, 2012).

3. Araucaria is from the University of Dundee. It is a tool for analysing, reconstructing and diagramming arguments. Users load the text of an argument into the tool and then highlight pieces of text to use as the basis of their argument: the premises and conclusions. These are added to the argument structure in diagrammatical form. This tool can also be used to manually evaluate the argument and each claim can be assigned a level of good / bad or strong / weak. Some of the argument schemes from Walton can also be used. Arguments can be used to search the Araucaria database which looks for text, structure and scheme matching. AraucariaDB is a repository of arguments that can be searched using Xpath (www.computing.dundee.ac.uk/staff/creed/araucaria/, 2012; Reed and Rowe, 2004).
4. Paramenides from the University of Liverpool uses Walton's sufficient condition-ing scheme for practical reasoning. It allows the public to post responses to pol-
itical policy proposals; the proposals are structured using an argument scheme
and the users can add in their views and evaluate the arguments
(http://www.csc.liv.ac.uk/~parmenides/, 2012; Cartwright and Atkinson, 2009).

5. ClaiMaker from the Open University uses the ScholOnto ontology, which sup-
ports basic reasoning, to model users’ views of academic research papers. They
have a database of claims made in papers which can be queried. The approach
is based upon that taken in Compendium a tool also from the Open University,
that enables concept mapping using the IBIS ontology (Uren et al., 2003).

Each specific language is designed for a specific tool and therefore is closely tied to
the tool. A standardised format for content or a common ontology or a way of mapping
between the current structures would enable data to be shared amongst sites.

2.4 Techniques for Identifying and Classifying Argumentation in Text

2.4.1 Classifying Arguments

Machine learning tools have been used to automatically classify arguments (Witten
and Frank, 2005; Rose et al., 2008). Machine learning algorithms require a good set of
features that strongly predict the classes and a feature space that is as small as possible
to reduce processing time. The following section looks at the different approaches,
algorithms and features that have been used to identify argumentation in text.

An example of an area where argument theory has been implemented is in the field
of Computer Supported Collaborative Work (CSCW). It is used to monitor and assist
in collaborative work and learning. Systems have been developed that are used by
students in online debates; these use analysis of the argument structure to make sure
students stick to the proposed topic areas and to identify interactions where prompts
can be used to assist students. These prompts encourage the students to ground and
warrant their claims and help them to come to productive conclusions.

Rose et al. (2008) use argument theory to classify text in the CSCW domain. In
their 2008 paper they describe how they identify features that can be extracted from
the text that are useful in predicting types of discourse interactions. A framework is
used to analyse the text based upon a classification scheme from Weinberger and Fischer (2006). This scheme contains two initial levels of argumentation: the micro-level of argumentation, an individual argument consisting of a claim supported by a ground with warrant possibly specified by a qualifier; and the macro-level of argumentation, argumentation sequences where single arguments are connected to create an argumentation pattern such as argument and counter-argument.

The specific items of the Weinberger and Fischer (2006) classification system used:

1. Epistemic activity (formal argument quality)
2. Micro level of argumentation
3. Macro level of argumentation
4. Social modes of co-construction
5. Reaction to previous contribution
6. Reaction to script (prompt)
7. Quoted text

The underlying aim of the work by Rose et al. (2008) was to automate the classification of textual data from computer supported collaborative learning tools into a coding scheme. This coding scheme was used to investigate how argumentative knowledge construction occurred in online discussions.

The work used the TagHelper Tool; this is a corpus analysis environment built on top of the Weka machine learning toolkit. The text analysed comprised 250 discussions from online discussion boards. Human annotators split each message from the discussion board into several spans of text which represent the different communication acts. The thread structure of the messages and the relationships between the text spans within each message were recorded for use by the machine learning algorithm. In total 1,250 coded text segments were extracted and classified by humans using the multidimensional coding scheme from Weinberger and Fischer (2006).

Rose et al. (2008) considered two tasks: which text features worked well to distinguish classes and an exploration of the suitability of different supervised machine learning algorithms for this task. The algorithms that were tested were Naive Bayes, Support Vector Machines and Decision Trees.

As a baseline they classified using SVM, Naive Bayes and Decision Trees algorithms and the top 100 unigram features. In addition to this they tested 7 different sets additional of features:

- Unigrams
2.4. Techniques for Identifying and Classifying Argumentation in Text

- Unigrams and line length
- Unigrams and part of speech (POS) bigrams
- Unigrams and bigrams
- Unigrams and punctuation
- Unigrams and stemming
- Unigrams and rare words removed
- Unigrams, line length, punctuation, and rare words removed

The classification was evaluated by determining the level of agreement between the two classifications of the data, automatic and manual. They found that in general using the SVM algorithm gave the most agreement.

There is an emphasis in the Rose et al. (2008) work on the selection of features that assist in classification. The features which are identified as useful are: punctuation, unigrams, POS bigrams, line length, containing a non-stop word, stemmed words, and rare words. They found that bigrams substantially reduced performance as each one was relatively rare. They found that with this data set the most predictive features are unigrams and punctuation.

This work also investigated if the context of the text would prove useful in classification, this was investigated through the analysis of the relationship of the text spans to each other and their depth in the thread. Sequential learning techniques were used to establish the context of the span of text. It was thought that this may particularly help with certain types of spans of text, for example a reaction to previous contribution cannot occur in an initial position, a specific discourse act is set up by those preceding it. The depth of the thread that the text span appears in is included as a feature by coding it using a Finite State machine feature which has two states, Q0 as initial state at the top of the message and Q1 when a quoted span of text is encountered. It remains in this state until a prompt is reached indicating the span of text that related to something someone else has said.

The results showed that context and depth features assisted in identifying Micro-level argumentation and depth features helped in identifying Macro-level argumentation and social modes of co-construction. It is suggested that this is because complex levels of argumentation like counter-examples are more likely to refer to already mentioned information whereas claims are not.

The set of features as described by Rose et al. (2008) was extended by Abbott et al. (2011) to assist in recognising disagreement in online discussion forums. The focus of this work was on the recognition of argument structure by identifying agreement.
and disagreement. The machine learning toolkit Weka was used with Naive-Bayes and JRip algorithms. In addition to a base set of n-gram features, several meta-post features were used such as, ‘poster id’ and time between posts. The experimentation with these meta post features gave a small increase in agreement between automatic classification and human annotation from 63% to 68%. This approach may be useful when investigating classifying text into argument structure: users tend quote each other in their comments to allow them to break down posts into individual points of argument which they can then respond to in isolation, and this meta-post information could be used to determine which is the quoted text and which is the new text and thereby differentiate between claims and counter claims.

Mishne and Glance (2006) discuss disputative comments, where users post comments that disagree with the previous comments with the intention of provoking debates and arguments. The disputative comments were identified using a decision tree built from manually classified comments. The most predictive features were found to be: the use of question marks early in the text, the number of comments in the thread, and the polarity of the first sentence of the comment. They also found that phrases that occur often in debates such as ‘I don’t think that’ or ‘you are wrong’ and the words ‘not’ and ‘but’ are strong features.

Somasundaran et al. (2008); Somasundaran and Wiebe (2010) found other specific cue phrases that are strong features for use in this type of classification. Discourse markers that are strongly associated with pragmatic functions can be used to predict the class of content, therefore useful features include the presence of a known marker such as ‘actually’, ‘because’, ‘but’, ‘i believe’, ‘i know’.

In general, it has been shown that it is difficult to vastly improve a baseline that uses only unigrams as features, although simple rules defined over the content can improve the features and therefore the classification. However, there is some indication that specific features, such as cue words or POS features, can be used to identify specific aspects of argumentation and in classifying the sentences into claims, grounds and qualifiers. These approaches will be explored to determine whether the classes taken from the Toumlin approach can be identified and extended.

Teufel and Moens (1997, 2002); Teufel et al. (1999) discuss how argumentative classification of extracted sentences can be used to summarize research articles and provide a degree of context for the extracted sentences that is missing in other summarization approaches. They extract sentences from the articles to form a summary but keep the overall rhetorical structure, rather than picking the highest rated sentences.
The rhetorical structure is used to aid extraction of text that describes specific aspects of a paper, whether the sentence represents amongst other things, a main goal of the article, a problem or a contrast to someone else’s work.

This research uses the CARS (Create a Research Space) model described by Swales (1990) for expressing how academics write papers. This model describes a framework of how, in general, authors of academic papers structure their writing by making specific moves such as pointing out weakness in others research. These moves can be identified using rhetorical markers but these markers mean different things depending on where they are in the text, the same marker might mean a different thing in a related work section as opposed to a conclusion.

They identify what they call meta comments in the text, these are phrases like ‘we have presented a method for’ to fill certain slots in the abstract (such as method). They use a Naive Bayes classifier to identify the sentences that fill a specific role in the summary to be created, such as a method role. This is done by extracting abstract-worthy sentences and the classification of the rhetorical role for that sentence.

They classify the text into basic and non-basic rhetorical categories which are taken from Swales CARS model (Swales, 1990). The basic categories are:

- Background, generally accepted statements
- Others, statements made by others
- Own, statements made by the author

In addition to this there is:

- Aim, which can be used to categorise the entire text
- Textual, providing information on the structure of the text
- Contrast, comparing the authors work with the work of others
- Basis, the work that the author builds upon.

The features that are used in the machine learning are quite complex and include features based on the structure, citation, syntactic, semantic and content aspects of the text. The features based on structure are created by determining the explicit structure of the text, such as position of a text span within the full text, the section, and the paragraph. Whether the text contains a citation (self or to others) is included as a feature. The syntactic features are based on tense and whether it contains modals, voice and negation. An example of the semantic features include the action type of the first sentence. Examples of content features are if the sentence contain key words as
defined by tf/idf or words from the title. The results from this work were variable but there is a strong indication that the approach of considering the context of the rhetorical marker aided in the classification of the text spans.

2.4.2 Identifying Arguments

2.4.2.1 Contrasting Idea Identification

Argumentation in text can be discovered through identifying sentences that contain contrasting ideas. A contrasting idea is identified as a sentence that summarises a claim and counter claim. This can be used to identify a contrast between two previous points.

Contrasting idea identification is used by Sndor and Vorndran (2009) to identify key sentences for use in highlighting key threads for reviews when pre-reviewing papers. They use natural language processing (NLP) through the rule-based dependency parser Xerox Incremental Parser (XIP) (Kaplan, 2006). XIP looks for sentences using grammar rules, these rules contain particular discourse functions (De Liddo and Buckingham Shum, 2010; De Liddo et al., 2012). The features of the key sentences are assigned by applying the concept-matching framework. It appears that this concept matching framework is made up of a database of sets of words which is manually curated to indicate specific aspects of the text, such as a summary. The words from this database are used as seeds and added to incrementally as data from a specific domain is processed, thereby reflecting the way points are discussed in a domain specific way.

Once parsed the key sentences are defined as those that describe the main points of a paper, thus allowing a reviewer to evaluate the flow of the ideas within the article. The sentences identified do not describe concepts but instead argue about concepts. They have found that a sentence may not present enough context to the user and suggest that a larger text span may be useful. They also find that they get a very large number of false positives which they suggest may be overcome by expanding the text span beyond the sentence level.

De Liddo et al. (2012) also use the XIP parser to aid in Contested Collective Intelligence, the theory of the processes involved in working with ideas that are not proven and can be contested. The contrasting ideas are extracted using XIP and used to analyse problems and aid critical thinking by exploring opposing or diverging opinions. Sentences in the text are labelled as contrasting ideas. Opposing ideas are matched and used for summarisation and contrast. An annotation and knowledge mapping sys-
2.4. Techniques for Identifying and Classifying Argumentation in Text

Cohere, is used to model the contested ideas and present them to users. The two approaches presented here suggest that using a rule based parser based on discourse functions would be useful for identifying sentences that contain argumentation. Unfortunately the rules used by XIP are not in the public domain, therefore current approaches to discourse analysis are investigated in the next section.

2.4.2.2 Rhetorical Structure Theory

Rhetorical structure theory involves the identification of the relations between spans of text and the clauses involved in the relationship. These relations are organised into a hierarchical structure (Mann and Thompson, 1987, 1988). Rhetorical semantic relations indicate how the spans of text are joined - the relationship between these spans. Schemes devised using this theory describes how spans of text can co-occur. There are differences in the way different groups precisely define these schemes and the terms used to define them, but the different schema can be aligned (Blair-Goldensohn et al., 2007).

Rhetorical semantic relations could be used to identify related spans of text that could be classed as using argumentation structure such as claims, counter claims, premises and conclusions. The relations are described in different ways dependent on the scheme followed, but generally the type of relations that would be useful in argument analysis include those described as CONTRAST, CAUSE-EXPLANATION-EVIDENCE, CONDITION and ELABORATION.

Marcu and Echihabi (2002) describe an approach to automatically define and locate these types of relation. CONTRAST and CAUSE-EXPLANATION-EVIDENCE are often signalled by cue phrases (Example 1) such as because or however or they can be expressed implicitly (Example 2). Detecting implicit expressions is more difficult and much of the work in this field is devoted to this.

1. Example 1: because of the expenses scandal there have been a spate of resignations in the government

2. Example 2: the government was once again beset by scandal. After several key resignations

Marcu and Echihabi (2002) look for the relations using defined patterns or rules. Models of the rhetorical semantic relations are defined and identified in the text using a Naive Bayes classifier. The models were learned automatically from a mas-
sive dataset. A large number of examples are identified from a corpus: for identifying CONTRAST sentences were extracted that were linked using the phrase ‘but’, CAUSE-EXPLANATION-EVIDENCE sentences were identified as those linked by ‘because’ or ‘thus’. They are able to improve the quality of training examples using topic segmentation and syntactic heuristics to filter out training examples which are invalid.

Sporleder and Lascarides (2008) also extend the work by identifying features based on POS and argument structure. They looked for linguistic cue phrases to identify, extract and label training data which is used in classification. This is thought to be suitable for dealing with relations that are difficult to determine using just cue phrases. Linguistic cue phrases are made up of textual features such as word stems, POS and tense. These are then used to expand upon the type of training data that can be used to classify the rhetorical semantic relations. They use Decision Trees, rather than Naive Bayes as the algorithm in the machine learning process. The features that are used in the classification are:

- Positional features: if the text is at the beginning or end of a paragraph
- Length of spans
- Lexical features: words which are not cue words but might indicate certain relations
- POS: particularly verbs, nouns and adjectives in spans
- Temporal features: tense and aspect measured by modality, aspect, voice and negation.
- Syntactic features: complexity measured by number of NPs, VPs, PPs, ADJP's and ADVP's
- Cohesion Features: measured by the number of pronouns, ellipses and if a text span ends in a VP ellipse.

Blair-Goldensohn et al. (2007) have refined the approach using parameter optimisation, topic segmentation and syntactic parsing. The parameter optimisation included (amongst others):

- Laplace smoothing
- Using a vocabulary of 6400 stems: replacing all others with a pseudo-token.
- Removing the stoplist: as the words frequently in a stop list are useful in the model
2.4. Techniques for Identifying and Classifying Argumentation in Text

- Using a minimum frequency cut off: any pair with a frequency of less that 4 is removed

Blair-Goldensohn et al. (2007) also reiterate that topic segmentation is important as related instances are unlikely to cross topic boundaries. They defined a topic boundary as at least three intervening sentences or a paragraph break. Their system does not seem to work significantly better but there is some evidence that it may perform better on a larger corpus.

2.4.2.3 Global Rhetorical Moves

As described in the classification section, Teufel and Moens (1997, 2002); Teufel et al. (1999) assigned sentences to argumentative roles to create and evaluate summaries of academic papers. The work is described as finding global rhetorical moves as opposed to rhetorical structure theory moves (Marcu, 1997) which are generally focused on local relationships.

Text spans are identified which characterise the articles they come from, such as: unfortunately this work does not solve problem X. The emphasis of the Teufel and Moens (1997, 2002); Teufel et al. (1999) work is that a text span has different meaning depending where it is in the text, for example in a future work section the example sentence would express some reservations about the work presented in the article and in an early section of the text it would indicate that this work is trying to solve problem X.

This approach could be applicable to argumentation. If in user generated content a text span such as the example above is presented towards the end of a user generated comment this type of phrase may indicate the user has expresses a reservation about the claim that they are making, what Toulmin would describe as a Qualifier, if it was towards the beginning of a comment it would indicate the following text is a counter argument to a previous claim. Teufel and Moens (1997, 2002); Teufel et al. (1999) restrict their work to academic writing as they think there are communication goals that are specific to this domain, but adapting this approach to various types of user generated content may provide interesting results.

2.4.2.4 Argument Discourse Markers

Discourse markers or cue phrases link the section they are introducing and the section that precedes. They are generally made up of conjunctions, adverbs and prepositional
phrases (Fraser, 1999). Argument discourse markers are words or phrases that link argumentative points. Tseronis (2011) describes three main approaches to describing argument markers: Geneva School, Argument within Language Theory and the Pragma-dialectical Approach.

The Geneva School approach describes the various relations in language, in a similar fashion to rhetorical structure theory. There are 3 main types of markers/connective, organisation markers, illocutionary function markers (the relations between acts) and interactive function markers. Among the interactive markers there are 3 sub groups: a relationship between the argument and the master act, a relationship between the counter-arguments and the master act and a reformulation of the acts that proceed.

Argument within Language Theory is a study of individual words and phrases. The words identified are argument connectors: these describe an argumentative function of a text span and change the potential of it either realising or de-realising the span. The adverbs and adjectives that accompany verbs and nouns change the argumentative force. These can therefore be used as cue words to identify argument connections and force.

The Pragma-dialectical Approach looks at the context beyond words and expressions that directly refer to the argument. It attempts to identify words and expressions that refer to any moves in the argument process (Tseronis, 2011). In a similar fashion to the work of Marcu and Echihabi (2002) the approach is to create a model of an ideal argument and annotate relevant units. Tseronis (2011) analysed text to identify markers of argumentative moves. The 4 units are indications of:

1. stand points (confrontation stage)
2. challenge to defend a stand point (opening stage)
3. argument schemes and related discussions (argumentation stage)
4. maintaining or withdrawing a standpoint of doubt (concluding stage)

Aspects of all of these approaches could be used to define rules that identify text spans which contain argumentative text and also do indicate the argumentative role that these text spans take such as claim or counter claim.

### 2.4.2.5 Additional Techniques

In addition to identifying and classifying specific text spans that contain argumentation points, other aspects of the user generated content could provide cues as to how to identify and structure the appropriate information. These techniques may include
2.4. Techniques for Identifying and Classifying Argumentation in Text

identification of the topics, an analysis of the quality of the content and using semantic analysis to determine the general opinion of the user. These techniques are discussed in more detail below.

Semantic analysis can be used to determine the opinions of the users from the text in the conversation. This could be used to model a specific user to identify likely opinions or to model a conversation and determine the movement of the argument positions over time. Sentiment analysis has been applied to text in many areas including blog posts, Twitter, and other social networking sites. Potthast and Becker (2010) summarised and visualised comments to give a general overview of the sentiment contained within Flickr and YouTube comments. It was determined that the semantic orientation of a word could be predicted by its association with known words from sentiment dictionaries. In a similar way the tool SentiWordNet assigns a positive, negative and objective score to words using a semi-supervised classifier and a training set from WordNet (Esuli and Sebastiani, 2006). Semantic analysis may be useful to distinguish opposing points in an argument such as claim and counter claim.

As mentioned in section 1.3 the approach Bal and Saint-Dizier (2009) have taken to opinion mining has been adapted to form the basic structure for the work proposed here. They investigate distinguishing between facts and opinions by looking at news stories to determine how convincing the arguments appear in supporting certain conclusions. The general procedure that they use in opinion mining is to determine the:

- Subjectivity (identifying the subjective and objective spans of text)
- Orientation (polarity) of the text spans
- Strength of the orientation (weak/strong)

The tasks that are required to do this are described as:

1. Identify topic (topic detection / topic clustering)
2. Identify relevant sentences for opinion analysis / summarization (classification)
3. Track opinion (sentiment scoring)
4. Identify opinion holders (identify authority claims)
5. Measure the strength of argument (evaluating expressions / opinion grading)
6. Produce a model of the discourse

Opinions are expressed in conjunction with evaluating expressions, which are the adverbs and adjectives. The strength of the opinion can be evaluated using these expressions, for example expressions of doubt could be ‘may be’, ‘possibly’, ‘probably’
and indicated by the presence of epistemic expressions, ‘I think’ and ‘I feel’. Claims of facts are grounded by an authority giving the evidence for the claim (similar to the claim, premise, grounds in Toulmin’s model). Therefore according to Bal and Saint-Dizier (2009) the claims / opinions can be graded from strongest to weakest:

1. Hypothesis statement (explains an observation)
2. Theory statement (widely believed explanation)
3. Assumptive statements (unprovable statements)
4. Value statements (based on personal belief)
5. Exaggerated statement (intended to sway readers)
6. Attitude statements (based on an implied belief system)

They use a polarity lexicon with expressions collected from a corpus and classified into positive and negative expressions. This is added to using a dictionary, thesaurus and WordNet.

Balasubramanyan et al. (2011) looked at comment polarity in political blog posts. Topics were identified which cause strong reaction within specific communities. They used sentiment polarity assigned to community specific models to summarise issues tackled. Using Support Vector Machine classification and supervised Latent Dirichlet Allocation topic models, it was found that emotional reactions are community specific. Those with opposing views would have different sentiment ratings for the same terms. Therefore it is possible to identify the view of an individual depending on the terms that they express a strong emotional response to, this could be a useful model to follow in modelling users’ opinions on certain topics.

Topic identification is useful in identifying subtopics within a discussion and could be used to label the discussions and sub discussions. TF-IDF can be used to determine how strongly terms are associated with specific topics (Manning et al., 2008), therefore past discussions can be used to help those newer to the discussions. Arguments can become cyclic, identifying the same small set of issues multiple times and never moving towards a solution. If topic identification is combined with a time order it could be used to determine if the argument is progressing and to see if the range of information covered is increasing.

The quality of user generated content is variable; therefore being able to classify the content according to the quality may be useful. Agichtein et al. (2008) studied the quality of answers in the Yahoo!Answer website. This website allows users to post and answer questions and vote on the best answers. Text analysis was conducted to assess
the quality of the answers. This included ideas from automatic essay grading (composition, style, accuracy and soundness) and used a range of features such as word length, measures of vocabulary irregularity, repetitiveness and measures of topicality, phrase frequency, punctuations and grammar. Readability was also used, text was analysed to determine the age group that it is suitable for. Measures of readability include the Flesch-Kincade formula and SMOG Grading, these analyse the number of words per sentence and number of syllables per word. They found that high quality user generated content, that which scores highly in the analysis described above correlated with a score for usefulness described by other means such as a rating system from other users or number of views.
Chapter 3

Approach

The aim of this work is to automatically identify and extract the structure of arguments from user generated content. This work aims to extract arguments from within a stream of conversations from user generated content sources such as Twitter, comments on news articles and others. In some cases there is therefore a step that precedes the argument analysis which is the identification of conversations. This type of problem has been tackled before in fields like opinion mining of product reviews. Therefore the approach that is taken in this work is inspired by Bal and Saint-Dizier (2009). The steps that they identify for processing product reviews are:

1. Find conversations on different topics

2. Identify relevant pieces of text

3. Classify into a structure as to define the relationship of these points

4. Present to users

The first three steps in the process are applicable to the identification of arguments in user generated content and will be investigated as part of this work. In addition, methods for presenting the extracted structure and interaction with that structure will also be investigated. It is also hoped that attempts can be made to model the users of the data so as to attempt to indicate useful points of interaction for those users with the different texts.
3.1 Work In Progress

An investigation into the current approach to argument analysis in user generated content has been conducted as specified in Chapter 2. In order to explore previous approaches in more detail, some of the experiments are re-implemented with new data. This work, and any novel experimentation, requires a data set - this is discussed in more detail in section 3.1.1. The analysis which has been conducted so far is then discussed. This includes: topic detection (section 3.1.2) and classification of text into different parts of a basic argumentation structure (section 3.1.3). Some conclusion are then drawn from the work so far in section 3.1.4.

3.1.1 Data Sets

The initial stages of this work will aim to identify arguments in a test data set. The data set that is being used initially is taken from Twitter. Twitter is a micro blogging site, where users can post text in 140 character segments. Jodi Schneider’s blog (jodischneider.com/blog, 2012) gives an example of argumentation structure in a single tweet in Twitter. This tweet is an example of a claim, evidence and grounds in less than 140 characters:

Difference between cakes and biscuits? When stale, cakes go hard, biscuits go soft. Hence Jaffa Cakes are cakes. (Was official EU ruling).

The above tweet represents several aspects of argumentation. The example below is an example of a claim without any evidence or grounds. It is taken from the London Riots data corpus.

Tigers running around London because someone broke into London zoo #WTF #LondonRiots

Text that is posted in Twitter is different from many other user generated content types as it is very short, it does not always use standard spelling or grammar and contains many specific conventions associated with this specific medium such as re-tweeting, hashtags and emoticons. The data is noisy and messy and covers a large number of parallel conversations all occurring at the same time. The data is freely available, in large volume and is in clearly structured short conversational pieces.

Ramage et al. (2010) and (Ritter et al., 2010) have analysed Twitter to categorise conversation and to identify specific dialogue acts using Latent Dirichlet Allocation.
3.1. Work In Progress

Table 3.1: Number of tweets annotated with argument classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Number of tweets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Claim Without Evidence</td>
<td>2117</td>
</tr>
<tr>
<td>Claim With Evidence</td>
<td>3644</td>
</tr>
<tr>
<td>Counter Claim Without Evidence</td>
<td>689</td>
</tr>
<tr>
<td>Counter Claim With Evidence</td>
<td>268</td>
</tr>
<tr>
<td>Implicit Request for Verification</td>
<td>579</td>
</tr>
<tr>
<td>Explicit Request for Verification</td>
<td>0</td>
</tr>
<tr>
<td>Comment</td>
<td>384</td>
</tr>
<tr>
<td>Other</td>
<td>13</td>
</tr>
<tr>
<td>Uncoded</td>
<td>35</td>
</tr>
</tbody>
</table>

They both confirm that while Twitter has very specific language and language constructions, the conversations conducted within this domain can be analysed in a conventional way.

A data set of hand annotated tweets has been acquired that describes events surrounding the London Riots in 2011. This data was originally used to create a visualisation of how rumours spread through Twitter during the London Riots (Procter et al., 2011). An example tweet is presented below, it is show in the JSON format, the content is the text contained in the body tuple.

```
{"author": "*****",
"body": "RT @******: #londonriots oh my god - reports of tigers roaming around Primrose Hill #londonzoobreakin http://t.co/j2DjbOZ",
"id": 683295, "influence": 72, "parent": 628335, "time": 1312836361,
"type": 2}
```

Each tweet was post-annotated with a code which expressed the type of argument within a simplistic structure, as can be seen in table 3.1. The codes were assigned by three human annotators from the University of Manchester. In total the data set contains 7729 tweets. The breakdown of the number of tweets annotated with each code is included in table 3.1.

As can be seen there are many more claims with and without evidence than other classes; this will be kept in mind when the results of experiments are interpreted. It
can also be seen that there are 35 tweets that are uncoded; it is unknown why this has occurred. These tweets were removed leaving a total of 7694 for further analysis.

It is a common practice within Twitter to re-tweet the tweets of others - this means to forward the tweet to your followers with attribution to the author of the original tweet. This can mean that a set of collected tweets can be repetitive with the same tweet repeated by many users. This kind of data may bias the machine learning algorithms, therefore the tweets were analysed to determine the amount of repetition and re-tweets in this dataset. Several tweets contained ‘RT’ indicating that the author had re-tweeted the text of another author and were acknowledging that they were re-tweeting. In addition to this there are authors that copy the tweets of other and do not attribute them (they omit the ‘RT’). Of the 7694 tweets there were 2786 individual tweets when all identical tweets were removed.

To provide some insight into the type of data to be analysed some examples of each class are included in table 3.2. As can be seen it is common to provide links to other websites especially in claims and counter claims with evidence and comments. Hashtags are liberally used, and re-tweets are common.

The human annotators assigned the tweets to specific topics which represented different events that took place in the riots. The numbers in each topic are in table 3.3, which shows both total number of tweets and unique tweets. It can be seen that some topics are more heavily represented that others. This may influence the results of the machine learning that is to be undertaken and will be discussed in another section.

This data will be used in the next stages of this work. It is intended that supervised machine learning techniques will be used to identify arguments in the test data set through learning from the labels that have been used in the annotation.

### 3.1.2 Topic Detection

In Twitter, many conversations occur within a stream of data which are overlaid and intertwined. It is therefore important to identify specific topics so that the comments on these topics can be structured together to provide an overview of the general conversation. Therefore the first investigation in the process is detecting the topics. As the data set is marked up with argument structure it is assumed that the entire 140 character string is relevant, therefore in this case text span identification is not required and will be investigated at a later point with a different data set. Once the topics have been identified, the classification of the tweets into an argument structure is investigated.
### Table 3.2: Example tweets - text from the ‘body’ section of the JSON format

#### Claim without evidence

RT @*******: a friend in London just told me it looks like the Army are gathering in the Bank area to go into action. #LondonRiots

RT @j********: Rumours going round that the army is on the street at Bank. #londonriots

#### Claim with evidence

RT @*******: The army arrive in Bank, East London http://yfrog.com/ki90044631j #LondonRiots - Here we go!!!

RT @*******: #LondonRiots the Army have arrived in Bank. http://t.co/FzG3p7e

#### Counter-claim without evidence

RT @*******: If the army have been deployed, which is very unlikely, Bank would be tactically flawed place to start #londonriots

RT @*******: Pic of ‘army assembling in Bank’ is not true. Old pic from Egypt apparently #mybad #londonriots

#### Counter-claim with evidence

RT @*******: This pic of “army tanks in Bank” going round is actually an image from Egypt. Scaremongering http://t.co/EcEzEqt #LondonRiots

RT @*******: Army is NOT in Bank. The photo is from Egypt: http://ow.ly/5YdIA #Londonriots

#### Implicit request for verification

RT @*******: Hi @skyruth or @Emmabarnett or @skynewsbreak are you hearing news of the army preparing at Bank? #londonriots

RT @*******: Is this image real, the army assembling in Bank. Why nothing on the news? http://t.co/HFVqC4b #londonriots

#### Comment

RT @*******: Army at bank rumours started by this picture: http://t.co/4A4YhB1 #LondonRiots

RT @*******: Oh, ffs, stop retweeting that Bank/army photo. #londonriots

#### Other

RT @*******: No lo puedo creer! :O RT @*******: London eye is burning #LondonRiots #prayforlondon http://t.co/LI53kWf

RT @*******: Vale, el London Eye NO EST ARDIENDO. #londonriots

#### None

RT @*******: #Birmingham children’s hospital has been attacked?? :( Hope this isn’t true! #riots

RT @*******: A children’s hospital, really???? how could anyone do that, I don’t understand. #birminghamriots
Chapter 3. Approach

<table>
<thead>
<tr>
<th>Topic</th>
<th>Number Tweets</th>
<th>Number Unique Tweets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Army at Bank</td>
<td>191</td>
<td>99</td>
</tr>
<tr>
<td>Rioting at Hospital</td>
<td>1628</td>
<td>625</td>
</tr>
<tr>
<td>London Eye on Fire</td>
<td>654</td>
<td>346</td>
</tr>
<tr>
<td>McDonalds</td>
<td>216</td>
<td>108</td>
</tr>
<tr>
<td>Miss Selfridge</td>
<td>3181</td>
<td>867</td>
</tr>
<tr>
<td>Police Beat Girl</td>
<td>888</td>
<td>305</td>
</tr>
<tr>
<td>Tiger escapes from London Zoo</td>
<td>936</td>
<td>436</td>
</tr>
</tbody>
</table>

The presentation of the structures to users will be investigated at a later date.

Initial experimentation has been conducted using the hand annotated Twitter set. The data has been split into 7 specific topics by human annotators, so it was investigated whether the topic detection could be replicated using an unsupervised machine learning technique.

The Natural Language Tool Kit (NLTK) was used with the K-means algorithm implementation (Bird et al., 2009). The features used were the most popular unigrams and bigrams (excluding a stop list of common terms). The K-means algorithm can be used in this implementation in two ways: by either specifying a number of clusters and allowing random initial locations as centroids for each cluster and moving this centroid over several iterations to achieve the smallest item to centroid distances or by setting the initial means. It was found that the most accurate clusters were produced using an initial mean calculated using seed words to represent each cluster. The words or phrases used were, army, hospital, london eye, mcdonalds, miss selfridge, police beat girl and zoo.

The seeds were used to identify tweets that were on specific topics. Tweets which contained the seed words were used to create a vector that represented an average tweet for this topic; this was then used as the initial point for the centroid for that topic. A RAND index was calculated for each cluster: this is a measure of the accuracy of each cluster and is measured by the percentage of decisions that were correct. Initial evaluation of this approach has shown that the topic can correctly be identified much of the
time (the minimum score was a RAND Index of 0.89) using unsupervised clustering for full results see table 3.4. It can also be seen from this table that when noise tweets are added (tweets unrelated to the topics searched for), 26294 extra, the accuracy improves. From the 7729 labelled tweets the system classified 829 wrongly initially and 408 wrongly once the noise tweets were added. Seeding this data originally with these words may not always be possible so alternate approaches need to be identified in order to initiate clusters.

The top ten most highly weighted features were identified for each cluster in the hope of providing an identifying topic label for the cluster; the results are presented in table 3.5.

Hierarchical clustering was investigated to see if the main topics split into sub topics. Hierarchical clustering was achieved by seeding the clustering as before and then splitting each cluster into two and then each of those clusters into two. This approach could be refined by identifying better methods for splitting the data beyond a binary split. An example of the type of result is seen in table 3.6.

In table 3.6 several terms are in italics. This is to indicate the terms that emerge during the hierarchical clustering which may express some form of argument structure. For example it is possible that confirm suggests that there is a claim substantiated with evidence. Photoshop and neither may suggest counter claim and evidence, report may suggest an initial claim without evidence. Further investigation into this approach is needed to confirm if this could be useful.

### Table 3.4: Topic Clustering Evaluation

<table>
<thead>
<tr>
<th>Group</th>
<th>7 Groups</th>
<th>With Noise</th>
</tr>
</thead>
<tbody>
<tr>
<td>police beat girl</td>
<td>0.99754</td>
<td>0.99888</td>
</tr>
<tr>
<td>miss selfridge</td>
<td>0.99017</td>
<td>0.99774</td>
</tr>
<tr>
<td>hospital</td>
<td>0.99780</td>
<td>0.99924</td>
</tr>
<tr>
<td>london eye</td>
<td>0.99547</td>
<td>0.99871</td>
</tr>
<tr>
<td>army</td>
<td>0.89287</td>
<td>0.99161</td>
</tr>
<tr>
<td>zoo</td>
<td>0.91163</td>
<td>0.99112</td>
</tr>
<tr>
<td>mcdonalds</td>
<td>1.00000</td>
<td>0.99273</td>
</tr>
</tbody>
</table>
Table 3.5: Topic Clustering Labels

<table>
<thead>
<tr>
<th>Group 0</th>
<th>polic, girl, old, tottenham, old girl, riot, year, http, start, beat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td>miss, selfridg, miss selfridg, fire, set, fire miss, set fire, http, manchesterriot, manchest</td>
</tr>
<tr>
<td>Group 2</td>
<td>children, hospit, children hospit, birmingham, birmingham-riot, birmingham children, attack, move, peopl, riot</td>
</tr>
<tr>
<td>Group 3</td>
<td>london, londonriot, eye, http, prayforlondon, fire, report, confirm, around, photo</td>
</tr>
<tr>
<td>Group 4</td>
<td>londonriot, zoo, tiger, london, http, report, londonzoo-breakin, roam, around, primros</td>
</tr>
<tr>
<td>Group 5</td>
<td>cook, mcdonald, tottenham, food, start, youth, loot, shop, riot, londonriot</td>
</tr>
</tbody>
</table>

Table 3.6: Hierarchical Clustering

<table>
<thead>
<tr>
<th>london, londonriot, eye, http, prayforlondon, fire, report, confirm, around, photo</th>
<th>londonriot, london, eye, london eye, http, eye londonriot, prayforlondon, londonriot prayfor-london, burn</th>
</tr>
</thead>
<tbody>
<tr>
<td>london, eye, londonriot, london eye, fire, eye fire, http, confirm, london londonriot, around</td>
<td>londonriot, eye, london eye, london eye, http, eye londonriot, prayforlondon, londonriot prayfor-london, burn</td>
</tr>
<tr>
<td>photoshop, london, londonriot, big, hellowooo, london eye, eye fire, neither, burn, ca</td>
<td>london, eye, londonriot, london eye, fire, eye fire, http, confirm, london londonriot, around</td>
</tr>
<tr>
<td>london, eye, londonriot, london eye, http, burn, eye londonriot, report, eye burn, prayforlon-don</td>
<td>londonriot, eye londonriot, londonriot prayforlondon, london, london eye, prayforlondon http, eye, http, london-riot londonriot</td>
</tr>
</tbody>
</table>
3.1.3 Argument Analysis and Classification

The aim of this work is to show that automated methods can be used to classify sections of text into argumentation types. Machine learning has been used in the past to automatically assign sections of text to a classification schemes (Witten and Frank, 2005; Rose et al., 2008).

Initially, supervised machine learning was used to classify the tweets. Rose et al. (2008) have investigated using supervised machine learning in order to classify text from online discussion forums in order to improve collaborative learning within that forum. This experimentation was used as a template thereby allowing a comparison of the results.

As described in more detail in the literature survey (in chapter 2) the Rose et al. (2008) experimentation compared the different machine learning algorithms for classifying text into a classification system from Weinberg and Fisher and they investigated which features were most useful in predicting the different classes.

This current work was done using the TagHelper Tool Set, as was the Rose et al. (2008) work. This is a tool which sits on top of the machine learning tool kit Weka. It allows users to extract features from text which are then used to create vectors which express a feature set for that text. In addition to unigram extraction (with or without a stop list and stemming), it allows the extraction of features based on punctuation, line length, bigrams, and part of speech (POS) bigrams. It also allows filtering of features below a certain threshold of frequency and there is the ability to create rules that group and/or map certain features.

The Twitter corpus (as described above in section 3.1.1) was used. Although these categories do not completely match those used in the Rose et al. (2008) work, described in more detail in section 2.4.1, the classes can be mapped as shown in table 3.7. Many of the codes from the Twitter scheme are combined to map to a single Weinberger and Fischer class (Weinberger and Fischer, 2006). In addition to this there are codes in the Weinberger and Fischer that are not present in the Twitter corpus.

The accuracy of the classification is evaluated using the Kappa statistic (Cohen, 1960) as used in the Rose et al. (2008) work. The Kappa statistic is intended to measure the agreement between two codings of the same data, in this case the human and the automatic classification. There is some disagreement over what constitutes an acceptable Kappa score, Rose et al. (2008) state that ‘A Kappa value of 0.4 is an acceptable level of agreement according to Fleiss and Cohen. However, that is substan-
Table 3.7: Alignment of coding schemes

<table>
<thead>
<tr>
<th>7 dimensions from Weinberger and Fischer</th>
<th>Coding Scheme used in the Twitter Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epistemic Activity</td>
<td>Comment</td>
</tr>
<tr>
<td></td>
<td>Other</td>
</tr>
<tr>
<td>Micro-level of argumentation</td>
<td>Claim without evidence</td>
</tr>
<tr>
<td></td>
<td>Claim with evidence</td>
</tr>
<tr>
<td>Macro-level of argumentation</td>
<td>Counter claim without evidence</td>
</tr>
<tr>
<td></td>
<td>Counter claim with evidence</td>
</tr>
<tr>
<td>Social modes of co-construction</td>
<td>Implicit request for verification</td>
</tr>
<tr>
<td></td>
<td>Explicit request for verification</td>
</tr>
<tr>
<td>Reaction to previous contribution</td>
<td>None</td>
</tr>
<tr>
<td>Reaction to script (prompt)</td>
<td>None</td>
</tr>
<tr>
<td>Quoted text</td>
<td>None</td>
</tr>
</tbody>
</table>

...tially lower than the more typical standard of 0.8 or at least 0.7, which is advocated by Krippendorf’. In this case, although the Kappa score is important for accessing the algorithms and features used, the comparison of the Kappa scores gained in these experiments and those found by Rose et al. (2008) indicate to what extent this approach can be reimplemented with different data.

### 3.1.3.1 Algorithms

The machine learning algorithms investigated were Naive Bayes (NB), Support Vector Machines (SVM) and Decision Trees (DT). These algorithms were investigated using a feature set which was made up of the top 100 most frequent unigrams for the Rose et al. (2008) work and top 82 unigrams for the Twitter work. As can be seen in table 3.8 the Rose et al. (2008) experimentation found that the SVM algorithm was the most effective algorithm at predicting the class whereas in this case for the majority of the time the better algorithm with the Twitter corpus is a DT.

The machine learning works well although it is supposed that the large number of re-tweets and the likelihood of a larger number of a certain class being about a certain topic may together be falsely increasing the score. Therefore two further experiments were conducted to ensure that these results were more indicative of results that may be
### Table 3.8: Classification of Online Discussion and Tweet Data Using Supervised Learning

<table>
<thead>
<tr>
<th>Discussion Forum</th>
<th>Algorithm</th>
<th>Kappa</th>
<th>Twitter Corpus Coding</th>
<th>Algorithm</th>
<th>Kappa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weinberger and Fischer Coding</td>
<td>SVM</td>
<td>0.60</td>
<td>Claim without evidence</td>
<td>DT</td>
<td>0.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Claim with evidence</td>
<td>DT</td>
<td>0.86</td>
</tr>
<tr>
<td>Micro-level of argumentation</td>
<td>SVM</td>
<td>0.70</td>
<td>Counter claim without evidence</td>
<td>DT</td>
<td>0.79</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Counter claim with evidence</td>
<td>DT</td>
<td>0.84</td>
</tr>
<tr>
<td>Macro-level of argumentation</td>
<td>SVM</td>
<td>0.48</td>
<td>Implicit request for verification</td>
<td>DT</td>
<td>0.47</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Explicit request for verification</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>Social modes of co-construction</td>
<td>SVM</td>
<td>0.53</td>
<td>Comment</td>
<td>DT</td>
<td>0.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Other</td>
<td>NB</td>
<td>0.63</td>
</tr>
</tbody>
</table>
obtained on other data sets. Firstly, all of the re-tweets were removed as discussed in the data analysis section so that only unique tweets remain (unique primary tweets and a single re-tweet of each tweet). As can be seen in table 3.9 (column titled no-repeats) results from the analysis gave kappa scores that were lower than the original tweet results as expected, but they are comparable with the Rose et al. (2008) results.

Secondly, it was presumed that the large number of tweets in particular classes that are on specific topics are skewing the results. Words that indicate the topics are being used to identify the class, when a class has many tweets from that topic for example the word hospital was a strong indication of the class implicit request for verification. Therefore it was important to run the experiments using non-topic specific words to avoid this skewing. To do this the most frequent 101 words from the stop list used in the the TagHelper Toolkit were used. The stop list words were used as this list was thought to be an example of likely frequent non-topic specific words. It is probable that better results would be found though the refinement of this list. A can be seen in table 3.9 (column titled Non-topic specific) that the results are are worse than, but close to, the no-repeats results particularly for the claim/counter claim with and without evidence.

These experiments suggest that the non-topic specific words are good features for classifying the data, in particular for the claims with and without evidence (0.73 and 0.80 Kappa respectively). The counter claims with and without evidence were not identified as accurately (0.45 and 0.52 Kappa respectively), although this is still higher than the minimum Kappa score for agreement presented by Fleiss and Cohen (1973) (0.4) but not the higher score presented by Krippendorf (0.7). The total number of tweets hand annotated as counter claims was much smaller than the number annotated as claims, the smaller training set may have influenced the accuracy and thereby the Kappa score.

3.1.3.2 Features

The second stage of the Rose et al. (2008) work was to investigate which features were the strongest predictors for argument classes. This was conducted using the features which can be extracted using the TagHelper Tools set as described above. Although the results for our machine learning show that with the Twitter data the Decision Tree algorithm was more effective, the SVM algorithm is used as this was the algorithm used in the results published in the Rose et al paper and therefore this enabled the comparison of results. The results for the DT correlated with the results for the SVM
Table 3.9: Classification of Online Discussion and Tweet Data Using Supervised Learning - Unigram Performance

<table>
<thead>
<tr>
<th>Discussion Forum</th>
<th></th>
<th>Twitter</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Full set</td>
<td>No repeats</td>
<td>Non-topic Specific</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Algo</td>
<td>Kappa</td>
<td>Algo</td>
<td>Kappa</td>
</tr>
<tr>
<td>Micro-level of argumentation</td>
<td>SVM 0.60</td>
<td>Claim w/o evidence</td>
<td>DT 0.84</td>
<td>SVM 0.74</td>
<td>SVM 0.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Claim with evidence</td>
<td>DT 0.86</td>
<td>SVM 0.81</td>
<td>DT 0.80</td>
</tr>
<tr>
<td>Macro-level of argumentation</td>
<td>SVM 0.70</td>
<td>Counter claim w/o evidence</td>
<td>DT 0.79</td>
<td>DT 0.58</td>
<td>DT 0.45</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Counter claim with evidence</td>
<td>DT 0.84</td>
<td>DT 0.57</td>
<td>SVM 0.52</td>
</tr>
<tr>
<td>Social modes of co-construction</td>
<td>SVM 0.48</td>
<td>Implicit request for verification</td>
<td>DT 0.47</td>
<td>DT 0.47</td>
<td>SVM 0.39</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Explicit request for verification</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Epistemic Activity</td>
<td>SVM 0.53</td>
<td>Comment</td>
<td>DT 0.49</td>
<td>NB 0.57</td>
<td>NB 0.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other</td>
<td>NB 0.63</td>
<td>NB 0.72</td>
<td>NB 0.54</td>
</tr>
</tbody>
</table>
Table 3.10: Feature Analysis (SVM)

<table>
<thead>
<tr>
<th>Features</th>
<th>Discussion Forum</th>
<th>Twitter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Kappa</td>
<td>Full set Kappa</td>
</tr>
<tr>
<td>Unigrams</td>
<td>0.48</td>
<td>0.68</td>
</tr>
<tr>
<td>Unigrams and Line Length</td>
<td>0.48</td>
<td>0.69</td>
</tr>
<tr>
<td>Unigrams and POS Bigrams</td>
<td>0.49</td>
<td>0.85</td>
</tr>
<tr>
<td>Unigrams and Bigrams</td>
<td>0.44</td>
<td>0.70</td>
</tr>
<tr>
<td>Unigrams and Punctuation</td>
<td>0.52</td>
<td>0.79</td>
</tr>
<tr>
<td>Unigrams and Stemming</td>
<td>0.49</td>
<td>n/a</td>
</tr>
<tr>
<td>Unigrams and No Stemming</td>
<td>n/a</td>
<td>0.66</td>
</tr>
<tr>
<td>Unigrams and Rare Words</td>
<td>0.48</td>
<td>0.68</td>
</tr>
<tr>
<td>Unigrams, Line Length, Punctuation and Rare Words</td>
<td>0.52</td>
<td>0.79</td>
</tr>
<tr>
<td>Unigrams and No Stop List</td>
<td>n/a</td>
<td>0.81</td>
</tr>
</tbody>
</table>

but were slightly better with all features. The results are shown in table 3.10.

It was decided to identify the best features for predicting argumentation. This includes the entire feature sets described in table 3.10. The features were analysed using the AttributeSelectClassifier algorithm which derived a list of the most influential features for each feature set, for example it gave the most influential unigrams, POS bigrams and punctuation. These were used to create a list of 46 attributes which were the most influential at predicting the argument classes; these results are compared in table 3.11.

### 3.1.3.3 Unsupervised Learning

Supervised machine learning requires a set of human annotated data in order to learn the classes. The classes are expressed as parameters which are used to classify unknown data. The field of user generated content is vast and the content type and topic is varied, so it is likely to require the production of annotated data for each different
Table 3.11: Best Feature Scores for Each Argument Class (using SVM classifier)

<table>
<thead>
<tr>
<th>Class</th>
<th>Unigrams (No Repeats, Non-topic Specific)</th>
<th>Unigrams + POS Bigrams</th>
<th>Handpicked Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Claim without evidence</td>
<td>0.79</td>
<td>0.79</td>
<td>0.82</td>
</tr>
<tr>
<td>Claim with evidence</td>
<td>0.73</td>
<td>0.79</td>
<td>0.67</td>
</tr>
<tr>
<td>Counter-claim without evidence</td>
<td>0.36</td>
<td>0.68</td>
<td>0.55</td>
</tr>
<tr>
<td>Counter-claim with evidence</td>
<td>0.52</td>
<td>0.75</td>
<td>0.42</td>
</tr>
<tr>
<td>Implicit request for verification</td>
<td>0.39</td>
<td>0.70</td>
<td>0.33</td>
</tr>
<tr>
<td>Comment</td>
<td>0.30</td>
<td>0.71</td>
<td>0.27</td>
</tr>
<tr>
<td>Other</td>
<td>0.36</td>
<td>0.45</td>
<td>0.18</td>
</tr>
</tbody>
</table>

content type. This task would be time consuming and likely be prohibitive in expanding the classification of data into argument types from other sources. Therefore it was decided to see if the features that are useful in classifying this data in a supervised sense could be used to split the data when used with unsupervised methods.

The 46 most influential features for supervised machine learning were used with an unsupervised method. Twenty percent of the original annotated Twitter data had not been used in the supervised experiments, this new data was used with the unsupervised method to ensure that the features selected in the previous experiment were not trained on this specific data. The hand annotations were used to evaluate if the approach worked by counting the numbers of each class in each cluster.

The Weka tool kit was used for the machine learning with the K-means clustering algorithm (the EM algorithm was also investigated but was found to be not as accurate). The 46 features were compared with other feature sets for comparison. The comparative feature sets were unigrams (the 101 words as specified in the list above for non-topic specific words), POS unigrams and POS bigrams.

As can be seen in table 3.12, although the hand picked attributes classify more than half of the instances incorrectly the results are an improvement on using unigrams, POS unigrams and POS bigrams.

Looking at the results expressed as a confusion matrix, where the tweets are shown it terms of how they were classified both by humans and automatically we can see
Table 3.12: Unsupervised Clustering

<table>
<thead>
<tr>
<th>Features</th>
<th>Number of Attributes</th>
<th>Incorrectly Clustered (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uni-grams</td>
<td>101</td>
<td>63.42</td>
</tr>
<tr>
<td>POS Uni-grams</td>
<td>39</td>
<td>72.07</td>
</tr>
<tr>
<td>POS Bi-grams</td>
<td>526</td>
<td>67.93</td>
</tr>
<tr>
<td>Hand Picked Features</td>
<td>46</td>
<td>53.15</td>
</tr>
</tbody>
</table>

Table 3.13: Confusion Matrix

<table>
<thead>
<tr>
<th>Class</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Claim without evidence</td>
<td>14</td>
<td>93</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>30</td>
<td>5</td>
</tr>
<tr>
<td>Claim with evidence</td>
<td>27</td>
<td>25</td>
<td>138</td>
<td>4</td>
<td>9</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Counter-claim without evidence</td>
<td>0</td>
<td>28</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>Counter-claim with evidence</td>
<td>0</td>
<td>9</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Implicit request for verification</td>
<td>3</td>
<td>25</td>
<td>23</td>
<td>7</td>
<td>2</td>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>Comment</td>
<td>4</td>
<td>16</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>16</td>
<td>7</td>
</tr>
<tr>
<td>Other</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

where the clusterer is making mistakes (see table 3.13). The numbers in the first row represent the automatically generated clusters, the classes are show in the first column. The cells highlighted are those which represent the cluster-class agreement, so the cluster for class ‘claim without evidence’ is 1. We can see that there is no cluster for the counter claim with evidence group as these items are clustered in the majority with the claim without evidence group, as are many of the counter claim without evidence. It is possible that sentiment analysis may assist in splitting these two groups, this is the next process that will be tried.

3.1.4 Conclusions from work so far

A Twitter data set was acquired and processed for use in the initial analysis.

Supervised machine learning was performed on the data and it was found that the supervised methods can learn to distinguish between the different types of argument
and that the results for use with this data are comparable with the Rose et al. (2008) work.

It was found that the Decision Tree algorithm performed better with this data set than the others tested, whereas Rose et al. (2008) found that a Support Vector Machine algorithm performed better.

A selection of features was identified as the most influential in classifying this data using supervised methods.

Unsupervised machine learning (clustering) performs fairly poorly when classifying this data set; the influential features selected from the supervised learning improved the results. Further work in this area is required. It is hoped that sentiment analysis may improve the clustering.
Chapter 4

Future Work

4.1 Approach

The project is split into four main sections which are represented in work packages:

1. Find conversations on different topics
2. Identify relevant pieces of text
3. Classify into a structure
4. Present to users

Each of these sections are designed as specific work packages and are composed of several tasks. As the project is experimental and likely to change in response to investigation the tasks within these work packages have a standard organisation and can be repeated cyclically to investigate different aspects of the project as seen in figure 4.2. These are:

1. Initial experimentation using a test dataset
2. Evaluation of that experimentation
3. Identification of issues and possible extensions
4. Investigation with a different data set
4.2 Plan

The interaction and the dependencies between the workpackages are shown in figure 4.2. The distribution over time is displayed in the Gantt Chart in figure 4.2.

**WP1 Data Set Identification** - The data that is processed in this project is integral to the success of the project. Initially, as discussed, the data that has been used is from Twitter. A second data set has been acquired from the Guardian Newspaper (www.guardian.co.uk, 2012), access has been provided to all of the comments from their website via an API. In addition, a data set that combines data from Twitter using the Open Repository Conference 2012 hashtag and abstracts from this conference has been made available for use in this project (http://or2012.ed.ac.uk/, 2012). Other data sources will be investigated as the project proceeds. This data is from multiple sites and therefore it has various ownership, access and permission for reuse conditions. These will be recorded for each data source and the various rules followed.

**WP2 Topic Identification** - The approach in the topic identification section is to use unsupervised clustering to identify sub-topics within a data set. Results of initial experimentation have been encouraging. The next phase in this work package is to add the ability to seed the clustering using initial high level data analysis, most likely using frequencies of uni/bigrams. The approach also needs to be implemented on a second data set.

**WP3 Text Span Identification** - The aim of this work package is to identify relevant text spans which can subsequently be classed into an argumentation structure. The initial experimentation will investigate if rule based approaches from Rhetorical Structure Theory can be implemented. This approach will be expanded using ideas from Contrasting Idea identification and Argument Discourse Markers. Identifying text spans has not been possible within the initial data set as the text is provided in very short initial spans. It is important therefore to use another data set such as the Guardian comments data in order to progress with this work package.

**WP4 Classification of text into an argument structure** - The initial approach used in this work package was to investigate if supervised machine learning could be used to assign a span of text to an argumentation class. This process was then used to identify which features were most useful for splitting this data. As user generated data comes
in many different formats with associated different types of structures, rules and specialisations these specific learned classes will probably not be extensible beyond the Twitter corpus. It was therefore decided to identify if an unsupervised approach which would be more flexible and therefore extendible to other data sets could be used. The most useful attributes were taken from the supervised approach and used to cluster data in an unsupervised manner. The results of this work showed some promise but were well below what would be required to classify the data. Therefore this approach will be extended through the investigation into hierarchical clustering which would more closely mirror the supervised decision tree algorithms which performed the best. Other complementary techniques such as sentiment analysis will be investigated.

**WP5 Presentation to users** - The approach taken in this workpackage will be to investigate approaches taken previously in argumentation mapping and visualisation systems. These systems allow users to create argument maps or semi-automated systems where the users annotate extracted arguments. It is proposed that current visualisation systems can be reused and extended to provide interaction with arguments extracted from user generated text. These systems will be evaluated to determine the most useful features and a prototype created.

**WP6 Integration** - The objective of this work is to produce a system which enables users to evaluate user generated content and add their own content to the conversation. Therefore the integration of the tools and techniques identified in work packages 2 - 5 is an important aspect of this work.

**WP7 Summative Evaluation** - The work package will cover the final aspects of evaluating the work conducted and the final tools and techniques produced. As the aim of the project is to identify whether the techniques discussed above can be used to assist users in reading, understanding comments and whereabout to add their own input therefore, the ideal situation would be to evaluate this work in conjunction with participants who have generated data. It is hoped that the system will be able to automatically generate appropriate locations for comments for participants. If participants can be identified then they will be asked to evaluate how well the system works. In addition, or as a primary focus if participants cannot be used, evaluation will be conducted using time slices to attempt to predict where a user will comment and then use data from a later time slice to see if they do.
WP8 - Project Management, Reporting and Publications - To ensure that this PhD work is completed in a timely fashion this work package includes tasks for the writing up of the progress reports and presentations as defined by the Universities policies. The write up of the final PhD thesis will be the focus of the majority of effort in the final six months of the project. In addition to this it is recognised that time needs to be allocated to the monitoring and writing of presentations, posters, abstracts and papers for relevant meetings and conferences.
4.2. Plan

Figure 4.2: Gantt Chart describing the Workpackages
Bibliography


www.guardian.co.uk (2012). Latest news, sport and comment from the guardian | the guardian. http://www.guardian.co.uk/.