Using Complex Argumentative Interactions to Reconstruct the Argumentative Structure of Large-Scale Debates

John Lawrence and Chris Reed
Centre for Argument Technology, University of Dundee, UK

Abstract

In this paper we consider the insights that can be gained by considering large scale argument networks and the complex interactions between their constituent propositions. We investigate metrics for analysing properties of these networks, illustrating these using a corpus of arguments taken from the 2016 US Presidential Debates. We present techniques for determining these features directly from natural language text and show that there is a strong correlation between these automatically identified features and the argumentative structure contained within the text. Finally, we combine these metrics with argument mining techniques and show how the identification of argumentative relations can be improved by considering the larger context in which they occur.

1 Introduction

Argument and debate form cornerstones of civilized society and of intellectual life. Processes of argumentation elect and run our governments, structure scientific endeavour and frame religious belief. Understanding the nature and structure of these argumentative processes has broad ranging applications including: supporting legal decision making (Palau and Moens, 2009); analysing product reviews to determine not just what opinions are being expressed, but why people hold those opinions (Wyner et al., 2012); opening up the complex debates in parliamentary records to a wider audience (Hirst and Feng, 2015); and providing in-depth, yet easily digestable, summaries of complex issues (Lawrence et al., 2016).

Argument Mining\textsuperscript{1} is the automatic identification of the argumentative structure contained within a piece of natural language text. By automatically identifying this structure and its associated premises and conclusions, we are able to tell not just what views are being expressed, but also why those particular views are held.

In this paper, we consider the insights that can be gained by considering large scale argument networks as a whole. We present two metrics, Centrality and Divisiveness which can be viewed as how important an issue is to the argument as a whole (how many other issues are connected to it), and how much an issue splits opinion (how many others issues are in conflict with it and the amount of support which the two sides have).

We first show how these metrics can be calculated from an annotated argument structure and then showing how they can be automatically approximated from the original text. We use this automatic approximation, reversing the original calculation, to determine the argumentative structure of un-annotated text. Finally, we combine this approach with existing argument mining techniques and show how the identification of properties of argumentative relations can be improved by considering the larger context in which these relations occurs.

2 Related Work

Despite the rich heritage of philosophical research in argumentation theory (van Eemeren et al., 2014; Chesñevar et al., 2006), the majority of argument mining techniques explored to date have focused on identifying specific facets of the argumentative structure rather than considering the complex network of interactions which occur in real-life debate. For example, existing approaches have considered, classifying sentences as argumentative or non-argumentative (Moens et al., 2007), classifi-
ing text spans as premises or conclusions (Palau and Moens, 2009), classifying the relations between specific sets of premises and their conclusion (Feng and Hirst, 2011), or classifying the different types of premise that can support a given conclusion (Park and Cardie, 2014).

The approach which we present in this paper considers large scale argument networks as a whole, looking at properties of argumentative text spans that are related to their role in the entire argumentative structure. In our automatic determination of Centrality and Divisiveness, we first construct a graph of semantic similarity between text spans and then use the Textrank algorithm (Mihalcea and Tarau, 2004) to determine those which are most central. For Divisiveness, we then look at the sentiment polarity of each text span compared to the rest of the corpus to measure how many others are in conflict with it and the amount of support which the two sides have. TextRank has been successfully applied to many natural language processing applications, including identifying those parts of a text which are argumentative (as opposed to those which are not) (Petasis and Karkaletsis, 2016).

Similarly, Wachsmuth et al. (2017) propose a model for determining the relevance of arguments using PageRank (Brin and Page, 1998). In this approach, the relevance of an argument’s conclusion is decided by what other arguments reuse it as a premise. These results are compared to an argument relevance benchmark dataset, manually annotated by seven experts. On this dataset, the PageRank scores are found to beat several intuitive baselines and correlate with human judgments of relevance.

Lawrence and Reed (2015) used semantic similarity to determine argumentative connections between text spans. The intuition being that if a proposition is similar to its predecessor then there exists some argumentative link between them, whereas if there is low similarity between a proposition and its predecessor, the author is going back to address a previously made point or starting a new topic. Using this method a precision of 0.72, and recall of 0.77 are recorded when comparing the resulting connections to a manual analysis, however it should be noted that what is being identified here is merely that an inference relationship exists between two propositions, and no indication is given as to the direction of this inference.

3 Data: The US 2016 Presidential Debate Corpus

The data which we use is taken from transcripts of the 2016 US presidential debates, along with a sampling of the online reaction to these debates. Specifically, the corpus consists of Argument Interchange Format (AIF) (Chesnèvar et al., 2006) analyses of the first general presidential head-to-head debate between Donald Trump and Hillary Clinton along with corresponding comments from threads on Reddit (reddit.com) dedicated to the debates as they were happening.

3.1 The Argument Interchange Format

The Argument Interchange Format is a popular standard for representing argument structures as graphs, founded upon philosophical research in argumentation theory (van Eemeren et al., 2014), implemented as a Semantic Web ontology, and recently extended to handle dialogical interaction (Reed et al., 2010). The AIF distinguishes information, I-nodes, from the schematic ways in which they are related, S-nodes. I-nodes represent propositional information contained in an argument, such as a conclusion, premise etc. A subset of I-nodes refers to propositional reports specifically about discourse events: these are L-nodes (locutions). S-nodes capture the application of schemes of three categories: argumentative, illocutionary and dialogical. Amongst argumentative patterns there are inferences or reasoning (RA-nodes), conflict (CA-nodes) and rephrase (MA-nodes). Dialogical transitions (TA-nodes) are schemes of interaction or protocol of a given dialogue game which determine possible relations between locutions. Illocutionary schemes (YA-nodes) are patterns of communicative intentions which speakers use to introduce propositional contents. These node types are summarised in Table 1.

3.2 Annotation

Analysis was performed using the OVA+ (Online Visualisation of Argument) analysis tool (Janier et al., 2014) to create a series of argument maps covering the entire televised debate along with online reaction consisting of sub-threads selected from the Reddit ‘megathreads’ created during the debate. Annotators were instructed to select sub-

2The full annotated corpus along with the original text is available at http://corpora.aifdb.org/US2016G1
threads based on three criteria (a) sub-threads must not be shorter than five turns; (b) sub-threads containing only jokes and wordplays are excluded; (c) technical and non-related threads are excluded. Details of the resulting corpora can be seen in Table 2 and a fragment of the analysed structure can be seen in Figure 1. The total number of RA and CA nodes is greater than the sum of these values for the TV and reddit corpora, this is due to additional connections linking these two corpora which appear in the combined corpus, but not in the individual copora. These connections mean that the total corpus forms a coherent whole where topics discussed in the televised debate are linked argumentatively to points made in the online discussion.

3.3 Inter-Annotator Agreement

Two analysts (A1, A2) completed analysis of televised debate; and a further two analysts (A3 and A4) worked on the reddit reaction. A subset of the dataset (approximately 10%) was randomly selected for duplicate annotation by two analysts and these sets were then used to calculate pairwise inter-annotator agreement. Measures of agreement were calculated using Cohen’s kappa $\kappa$ (Cohen, 1960) ($\kappa = 0.55$) and the Combined Argument Similarity Score version of $\kappa$, CASS-$\kappa$ (Duthie et al., 2016), which refines Cohen’s $\kappa$ to avoid over-penalizing for segmentation differences (CASS-$\kappa = 0.71$). In the former case, Cohen’s $\kappa$ is difficult to apply directly, because it assumes that the items being categorized are fixed — in this case, the items being categorized are segments, whereas analysts may differ on segmentation boundaries.

4 Large Scale Argument Graph Properties

The argument graphs described in the previous section allow us to look at the structure of the debate as a whole rather than focusing on the properties of individual relations between propositions. In this section we look at two measures, Centrality and Divisiveness, that individual propositions (I-nodes) exhibit which can only be interpreted when considering the broader context in which they occur.

Whilst there are certainly other measures that could be applied to an argument graph highlighting interesting features of the arguments being made, we have selected these two metrics as they can both be calculated as properties of the argument graph and approximations can be determined directly from the original text. In Section 5, we describe methods to determine these approximations directly from the original text. By first calculating them directly we can then reverse the process of determining them from the argumentative structure, cutting the manual analysis out of the loop and allowing us to determine the argumentative structure directly. In Section 6, we look at how this approach can be used to improve the accuracy of extracting the full argumentative structure directly from un-annotated text.

4.1 Centrality

Central issues are those that play a particularly important role in the argumentative structure. For example, in Figure 1, we can see that the node “CLINTON knows how to really work to get new jobs...” is intuitively more central to the dialogue, being the point which all of the others are responding to, than the node “CLINTON’s husband signed NAFTA...”. In order to calculate centrality scores for each

<table>
<thead>
<tr>
<th>Node</th>
<th>Component</th>
<th>Category</th>
<th>Node</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>I-node</td>
<td>Information (propositional contents)</td>
<td>I-node but not L-node contents of locutions</td>
<td>L-node</td>
<td>.locutions</td>
</tr>
<tr>
<td>S-node</td>
<td>Schemes (relations between contents)</td>
<td>Argument schemes</td>
<td>RA</td>
<td>inference</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CA</td>
<td>conflict</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>MA</td>
<td>rephrase</td>
</tr>
<tr>
<td></td>
<td>Ilocutionary schemes</td>
<td>TA</td>
<td>illocutionary connections</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dialogue schemes</td>
<td>TA</td>
<td>transitions</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Types and sub-types of nodes in the AIF standard and components of analysed argument data, and the categories of schemes.
To calculate eigencentrality, we start by defining a directed graph $G = (V, E)$, where vertices $V$ are propositions, locations, or relations between propositions, and edges $E$ represent support or conflict. We can then define eigencentrality over $G$ as the measure that captures the importance of each proposition in the debate.

First, we consider the complete AIF structure as a directed graph, $G = (V, E)$, in which vertices ($V$) are either propositions, locations, or relations between propositions; and those relations are either support, conflict, rephrase, illocution, or transition, captured by a function $R$ which maps $V \rightarrow \{prop, loc, support, conflict, rephrase, illocution, transition\}$ and edges exist between them $E \subseteq V \times V$.

From this we build the subgraph corresponding only to vertices connected by support or conflict relationships, which we call $G_{l} = (V_{l}, E_{l})$, where $V_{l} = \{v \in V : R(V) \in \{support, conflict\}\}$ and $\forall v_{l} \in V_{l}$ if $(v_{l}, v') \in E$, then $(v_{l}, v') \in E_{l}$ and if $(v', v_{l}) \in E$, then $(v', v_{l}) \in E_{l}$. We can then define eigencentrality over $G_{l}$ as in Equation 1, where $\lambda$ is a constant representing the greatest eigenvalue for which a non-zero eigenvector solution exists.

$$Central(v) = \frac{1}{\lambda} \sum_{v' \in V_{l}} \text{Central}(v') \text{ s.t. } (v, v') \in E_{l}$$

(1)

This results in a centrality score for each proposition, from which we can rank the propositions by how central they are to the debate. The top four ranked central propositions are listed below:

- **CLINTON is hinting at tax incentives**

   It is encouraging that these issues all concern the economy, which Pew Research identified as the single most important issue to voters (with 84% of voters ranking it as “very important”) in the 2016 US presidential elections.

4.2 Divisiveness

Divisive issues are those that split opinion and which have points both supporting and attacking them (Konat et al., 2016). Looking again at Figure 1, we can see that the node “CLINTON knows how to really work to get new jobs...” is not only central, but also divisive, with both incoming support and conflict. At the opposite end of the scale, the node “CLINTON has been a secretary of state”, is not divisive; such factual statements are unlikely to be disputed by anyone on either side of the debate.

The Divisiveness of an issue measures how many others are in conflict with it and the amount of support which the two sides have. By this measure, every proposition $v_{2}$ which is in conflict with $v$ (i.e. for which there is an edge either outgoing from $v$ through a conflict $v_{c}$ to $v_{2}$, or in the other direction, or both) is assessed for its support in comparison to that for $v$ and the sum over all such $v_{2}$ yields an overall measure of Divisiveness as shown in Equation 2, in which $|v|_{R(v)}^{\text{in}}$ refers to the in order of vertex $v$ where constraint $R(v)$ is met.

$$\text{Divisiveness} = \sum_{v_{2}} \text{Central}(v_{2})$$

Again we list the top four ranked divisive issues below, and it is certainly easy to see how such statements on the character of the candidates, the validity of their claims and controversial issues such as gun control could easily divide those commenting on the debate:

- **TRUMP settled that lawsuit with no admission of guilt**

- **I still support hand guns though**

---

Table 2: US 2016 General Presidential Debate Corpora statistics, listing word counts, propositions (I-nodes), supporting arguments (RA-nodes) and conflicts (CA-nodes).

<table>
<thead>
<tr>
<th></th>
<th>Words</th>
<th>I-nodes</th>
<th>RA-nodes</th>
<th>CA-nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Televised Debate</td>
<td>17,190</td>
<td>1,473</td>
<td>505</td>
<td>79</td>
</tr>
<tr>
<td>Reddit Reaction</td>
<td>12,694</td>
<td>1,279</td>
<td>377</td>
<td>242</td>
</tr>
<tr>
<td><strong>Total (US2016G1 Corpus)</strong></td>
<td><strong>29,884</strong></td>
<td><strong>2,752</strong></td>
<td><strong>901</strong></td>
<td><strong>347</strong></td>
</tr>
</tbody>
</table>

---

4.2 Divisiveness

Divisive issues are those that split opinion and which have points both supporting and attacking them (Konat et al., 2016). Looking again at Figure 1, we can see that the node “CLINTON knows how to really work to get new jobs...” is not only central, but also divisive, with both incoming support and conflict. At the opposite end of the scale, the node “CLINTON has been a secretary of state”, is not divisive; such factual statements are unlikely to be disputed by anyone on either side of the debate.

The Divisiveness of an issue measures how many others are in conflict with it and the amount of support which the two sides have. By this measure, every proposition $v_{2}$ which is in conflict with $v$ (i.e. for which there is an edge either outgoing from $v$ through a conflict $v_{c}$ to $v_{2}$, or in the other direction, or both) is assessed for its support in comparison to that for $v$ and the sum over all such $v_{2}$ yields an overall measure of Divisiveness as shown in Equation 2, in which $|v|_{R(v)}^{\text{in}}$ refers to the in order of vertex $v$ where constraint $R(v)$ is met.

$$\text{Divisiveness} = \sum_{v_{2}} \text{Central}(v_{2})$$

Again we list the top four ranked divisive issues below, and it is certainly easy to see how such statements on the character of the candidates, the validity of their claims and controversial issues such as gun control could easily divide those commenting on the debate:

- **TRUMP settled that lawsuit with no admission of guilt**

- **I still support hand guns though**
\[ \text{Divisive}(v) = \sum_{v_2 \in V \ s.t. \ ((v_2, v), (v, v_2)) \in E \lor (v, v_2) \in E} |v_2| \text{in} \ R(v') = \text{support} \times |v_2| \text{in} \ R(v') = \text{support} \]

- people have looked at both of our plans, have concluded that CLINTON’s would create 10 million jobs and TRUMP’s would lose us 3.5 million jobs
- CLINTON didn’t realize coming off as a snarky teenager isn’t a good look either

5 Automating the Identification of Large Scale Argument Graph Properties

In this section we investigate techniques to automatically rank text fragments by their centrality and divisiveness with no prior knowledge of the argumentative structure contained within the text. In each case, we take the manually segmented propositions from our corpus and apply techniques to rank these, we then compare the resulting rankings to the ranking determined from the manually analysed argument structures as described in Section 4.

5.1 Automatic Identification of Centrality

In order to calculate centrality automatically, we first hypothesise that propositions (I-nodes) that are connected by relations of either support or attack in an AIF graph will have a higher semantic similarity than those which have no argumentative connection. We can again see an example of this in Figure 1, where the node “CLINTON knows how
to really work to get new jobs and to get exports
that..." is connected via support and attack rela-
tions to nodes whose propositional contents are all
related to jobs or exports. The remaining nodes
in this example fragment all discuss more distant
concepts, such as Clinton’s experience.

We consider a range of methods for determin-
ing semantic similarity and in each case use these
as the edge weights in an automatically generated
similarity graph. We can then consider centrality
to be determined by high similarity to the greatest
number of other nodes. As such, we can use
TextRank (Mihalcea and Tarau, 2004) to produce a
centrality ranking directly from the text and com-
pare this to the ranking obtained from the argu-
mentative structure.

The first approach to determining similarity that
we consider is calculated as the number of com-
mon words between the two propositions, based
on the method proposed by Mihalcea and Tarau
(2004) for ranking sentences. Formally, given two
propositions $P_i$ and $P_j$, with a proposition being
represented by the set of $N_i$ words that appear in
the proposition $P_i = w_1^i, w_2^i, ..., w_{N_i}^i$, the simi-
arity of $P_i$ and $P_j$ is defined as:

$$\text{Similarity}(P_i, P_j) = \frac{|\{w_k | w_k \in P_i \land w_k \in P_j\}|}{\log(|P_i|) + \log(|P_j|)}$$

(3)

Whilst this approach is sufficient to determine
similarity in the example discussed above, it is re-
liant on the exact same words appearing in each
proposition. In order to allow for the use of syn-
onyms and related terms in the dialogue, we con-
sider several further measures of semantic simi-
larity.

The first of these approaches uses WordNet
(Miller, 1995) to replace the binary matching of
words in the method above with the distance be-
tween the synsets of each word. This value is in-
versely proportional to the number of nodes along
the shortest path between the synsets. The short-
est possible path occurs when the two synsets are
the same, in which case the length is 1, giving the
same result for exactly matching words.

We also tested two further methods of deter-
ming semantic similarity which have both been
shown to perform robustly when using models
trained on large external corpora (Lau and Bal-
win, 2016).

The first of these approaches uses word2vec
(Mikolov et al., 2013), an efficient neural ap-
proach to learning high-quality embeddings for
words. Due to the relatively small size of our train-
ing dataset, we used pre-trained skip-gram vectors
trained on part of the Google News dataset. This
model contains 300-dimensional vectors for 3 mil-
lion words and phrases obtained using a simple
data-driven approach described in Mikolov et al.
(2013).

To determine similarity between propositions,
we located the centroid of the word embeddings
for each by averaging the word2vec vectors for the
individual words in the proposition, and then cal-
culating the cosine similarity between centroids to
represent the proposition similarity.

The final approach which we implemented uses
a doc2vec (Le and Mikolov, 2014) distributed bag
of words (dbow) model to represent every propo-
sition as a vector with 300 dimensions. Again, we
then calculated the cosine similarity between vec-
tors to represent the proposition similarity.

For each of the methods described above, we
applied the ranking algorithm to give an ordered
list of propositions, we then compared the ranking
obtained by each to the centrality ranking calcu-
lated for the manually annotated argument struc-
ture, as described in Section 4, by calculating
the Kendall rank correlation coefficient (Kendall,
1938). The results for each method are shown in
Table 3. In each case the results show a correla-
tion between the rankings ($p < 0.05$) suggesting
that all of these methods are able to approximate
the centrality of propositions in the argumentative
structure. In Section 6 we explore these results fur-
ther and show that these approximations are in all
cases sufficient to improve the automatic extrac-
tion of the argumentative structure directly from
the original text.

5.2 Automatic Identification of Divisiveness

Whilst divisiveness is a related concept to centra-
ility, it is more challenging to determine directly
from the text, as we need to not only locate those
nodes that are most discussed, but also to limit this
to those which are involved in conflict relations.

Here we implement a method of determining
conflict relations using SentiWordNet, a lexical
resource for opinion mining. SentiWordNet as-
signs a triple of polarity scores to each synset of

---

5https://code.google.com/archive/p/word2vec/

6http://sentiwordnet.isti.cnr.it/
WordNet, a positivity, negativity and objectivity score. The sum of these scores is always 1. For example, the triple (1, 0, 0) (positivity, negativity, objectivity) is assigned to the synset of the word “good”.

Each proposition (I-node), is split into words and each word is stemmed and tagged, and stop words are removed. If a stemmed word belongs to one of the word classes “adjective”, “verb” or “noun”, its polarity scores are looked up in SentiWordNet. Where a word has multiple synsets, each of the polarity scores for that word are averaged across all of its synsets. The scores of all words within a sentence are then summed and divided by the number of words with scores to give a resulting triple of \{positivity, negativity, objectivity\} values for each proposition.

Having calculated the polarity triples for each proposition, we are then able to calculate the difference in polarity between two propositions, \( P_i \) and \( P_j \) as in equation 4.

\[
\text{We compute these differences in polarity for each pair of propositions in the corpus and then, for each of the methods of determining similarity discussed in the previous Subsection, multiply the similarity scores by the polarity difference to obtain a value representing the likelihood of conflict between the two. Finally for each proposition, we mirror the method of computing divisiveness from the argument graph. To do this, we look at each proposition, and take the sum of the centrality scores multiplied by the conflict value for each other proposition.}
\]

Following this approach for each method of determining similarity again gives us a ranking which we can then compare to the divisiveness ranking calculated for the manually annotated argument structure, as described in Section 4. For each approach, we again calculate the Kendall rank correlation coefficient. These results are shown in Table 4. We can see from these results that whilst there is still a positive correlation between the rankings, these are substantially less significant than those obtained for the centrality rankings. In the next Section we investigate whether these values are sufficient to have a positive impact on the argument mining task.

### Table 3: The Kendall rank correlation coefficient (\( \tau \)) for the rankings determined using TextRank for each method of determining semantic similarity compared to the Centrality ranking obtained from the manually annotated argument structure.

<table>
<thead>
<tr>
<th>Similarity Method</th>
<th>Kendall ( \tau )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common words</td>
<td>0.524</td>
</tr>
<tr>
<td>WordNet Synsets</td>
<td>0.656</td>
</tr>
<tr>
<td>Word2vec</td>
<td>0.618</td>
</tr>
<tr>
<td>Doc2vec</td>
<td>0.620</td>
</tr>
</tbody>
</table>

### Table 4: The Kendall rank correlation coefficient (\( \tau \)) for the Divisiveness rankings for each method of determining semantic similarity compared to the Divisiveness ranking obtained from the manually annotated argument structure.

<table>
<thead>
<tr>
<th>Similarity Method</th>
<th>Kendall ( \tau )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common words</td>
<td>0.197</td>
</tr>
<tr>
<td>WordNet Synsets</td>
<td>0.284</td>
</tr>
<tr>
<td>Word2vec</td>
<td>0.167</td>
</tr>
<tr>
<td>Doc2vec</td>
<td>0.133</td>
</tr>
</tbody>
</table>

### 6 Validation: Applying Automatically Identified Centrality and Divisiveness Scores to Argument Mining

Our final step is to validate both our concepts of centrality and divisiveness as calculated from annotated argument structures and our methods of calculating these same metrics directly from unannotated text. To do this, we adapt the “Topical Similarity” argument mining technique presented in (Lawrence et al., 2014), where it is assumed firstly that the argument structure to be determined can be represented as a tree, and secondly, that this tree is generated depth first. That is, the conclusion is given first and then a line of reasoning is followed supporting this conclusion. Once that line of reasoning is exhausted, the argument moves back up the tree to support one of the previously made points. If the current point is not related to any of those made previously, then it is assumed to be disconnected and possibly the start of a new topic.

Based on these assumptions the argumentative structure is determined by looking at how similar each proposition is to its predecessor. If they are sufficiently similar, it is assumed that they are connected and that the line of reasoning is being followed. If they are not sufficiently similar, then it is first considered whether we are moving back up the tree, and the current proposition is compared to
\[
Polarity(P_i, P_j) = \frac{|\text{positivity}(P_i) - \text{positivity}(P_j)| + |\text{negativity}(P_i) - \text{negativity}(P_j)|}{2} \tag{4}
\]

all of those statements made previously and connected to the most similar previous point. Finally, if the current point is not related to any of those made previously, then it is assumed to be disconnected from the existing structure. This process is illustrated in Figure 2.

Lawrence et al. perform these comparisons using a Latent Dirichlet Allocation (LDA) topic model. In our case, however, the argument structures we are working with are from much shorter pieces of text and as such generating LDA topic models from them is not feasible. To achieve the same task, we use the same semantic similarity measures described in Section 5. As in (Lawrence et al., 2014), the threshold required for two propositions to be considered sufficiently similar can be adjusted, altering the output structure, with a lower threshold giving more direct connections and a higher threshold greater branching and more unconnected components.

We first carried out this process for each method of computing semantic similarity using the same methodology as Lawrence et al. We then adapted Step 2 from Figure 2 by considering all of the previous propositions as potential candidate structures and, having produced these candidate structures calculated the Centrality and Divisiveness rankings for each structure as described in Section 4. Finally we computed the Kendall rank correlation coefficient comparing the centrality ranking of each candidate structure to the ranking computed only using similarity (as described in Section 5) and selected the structure which maximised the rank correlation.

Table 5 shows the precision, recall and F1-scores for automatically determining connections in the argumentative structure using each semantic similarity measure combined with maximising the rank correlations for centrality and divisiveness. We can see from these results that maximising divisiveness results in small increases in accuracy, and in all cases maximising centrality results in increased accuracy in determining connections, with increases of 0.03–0.05 in F1-score demonstrated for all the methods considered.

### 7 Conclusion

In this paper we have presented two metrics, Centrality and Divisiveness, for describing the nature of propositions and their context within a large scale argumentative structure. We have shown how these metrics can be calculated from annotated argument structures and produced reliable estimations of these metrics that can be extracted directly from un-annotated text, with strong positive correlations between both rankings.

Finally, we have shown how these metrics can be used to improve the accuracy of existing argument mining techniques. By broadening the focus of argument mining from specific facets, such as classifying as premise or conclusion, to look at features of the argumentative structure as a whole, we have presented an approach which can improve argument mining results either as a feature of existing techniques or as a part of a more robust ensemble technique such as that presented in (Lawrence and Reed, 2015).

<table>
<thead>
<tr>
<th>Similarity Method</th>
<th>p</th>
<th>r</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common words</td>
<td>0.66</td>
<td>0.51</td>
<td>0.58</td>
</tr>
<tr>
<td>+ Max Centrality</td>
<td><strong>0.68</strong></td>
<td><strong>0.55</strong></td>
<td><strong>0.61</strong></td>
</tr>
<tr>
<td>+ Max Divisiveness</td>
<td>0.67</td>
<td>0.51</td>
<td>0.58</td>
</tr>
<tr>
<td>WordNet Synsets</td>
<td>0.75</td>
<td>0.63</td>
<td>0.68</td>
</tr>
<tr>
<td>+ Max Centrality</td>
<td><strong>0.81</strong></td>
<td><strong>0.64</strong></td>
<td><strong>0.72</strong></td>
</tr>
<tr>
<td>+ Max Divisiveness</td>
<td>0.77</td>
<td>0.63</td>
<td>0.69</td>
</tr>
<tr>
<td>Word2vec</td>
<td>0.72</td>
<td>0.74</td>
<td>0.73</td>
</tr>
<tr>
<td>+ Max Centrality</td>
<td><strong>0.78</strong></td>
<td><strong>0.78</strong></td>
<td><strong>0.78</strong></td>
</tr>
<tr>
<td>+ Max Divisiveness</td>
<td>0.72</td>
<td>0.77</td>
<td>0.74</td>
</tr>
<tr>
<td>Doc2vec</td>
<td>0.67</td>
<td>0.66</td>
<td>0.66</td>
</tr>
<tr>
<td>+ Max Centrality</td>
<td><strong>0.73</strong></td>
<td><strong>0.70</strong></td>
<td><strong>0.71</strong></td>
</tr>
<tr>
<td>+ Max Divisiveness</td>
<td>0.69</td>
<td>0.67</td>
<td>0.68</td>
</tr>
</tbody>
</table>

Table 5: Precision, recall and F1-scores for automatically determining connections in the argumentative structure using each semantic similarity measure combined with Centrality and Divisiveness.

### Acknowledgments

This work was funded in part by EPSRC in the UK under grant EP/N014871/1.
Step 1: The similarity of a new proposition to its immediate predecessor is calculated. If the new proposition is sufficiently similar, this is viewed as a continuation of the previous line of reasoning and the two are connected.

Step 2: If the new proposition is not sufficiently similar to its immediate predecessor, the similarity to all previous propositions is calculated. The most similar previous proposition is then selected and, if it is sufficiently similar to the new proposition, a connection is made.

Step 3: If the new proposition is not sufficiently similar to any of the previous propositions, it is viewed as the start of a new line of reasoning, disconnected to the existing argument structure.

Figure 2: The steps involved in determining how the argument structure is connected using the “Topical Similarity” argument mining technique presented in (Lawrence et al., 2014).

References


First Workshop on Argumentation Mining. Association for Computational Linguistics, Baltimore, Maryland, pages 79–87.


