An Abstract Argumentation-Based Approach to Automatic Extractive Text Summarization

Stefano Ferilli(✉) and Andrea Pazienza

Dipartimento di Informatica, Università di Bari, Bari, Italy
{stefano.ferilli,andrea.pazienza}@uniba.it

Abstract. Sentence-based extractive summarization aims at automatically generating shorter versions of texts by extracting from them the minimal set of sentences that are necessary and sufficient to cover their content. Providing effective solutions to this task would allow the users of Digital Libraries to save time in selecting documents that may be appropriate for satisfying their information needs or for supporting their decision-making tasks. This paper proposes an approach, based on abstract argumentation, to select the sentences in a text that are to be included in its summary. The proposed approach obtained interesting experimental results on the English subset of the benchmark MultiLing 2015 dataset.

Keywords: Text summarization · Digital libraries
Abstract argumentation

1 Introduction

Text summarization aims at automatically creating a shorter version of (a set of) text document(s). Abstractive techniques [1] produce summaries that may contain sentences not present in the input document(s). Extractive techniques [14] select a subset of sentences from the input document(s). An accurate extractive summarization method must optimize two important properties [22]: *coverage*, expressing how much the method is able to cover a sufficient amount of topics from the original text, and *diversity*, which refers to the capability of the method of generating non-redundant information in the summary. Graph-based methods for automatic text summarization have provided encouraging results. Nodes in the graph are sentences in the input document(s), and weighted edges are placed whenever a node/sentence refers to another. Weights are used to generate the scores of sentences.

Argumentation is an inferential strategy that aims at selecting reliable items in a set of conflicting claims (the *arguments*). It has been a very active topic in Artificial Intelligence since more than two decades now. Specifically, the Abstract Argumentation Frameworks [6] work on graphs in which nodes represent arguments, and edges represent attack (or, sometimes, support) relationships among arguments. Several non-monotonic reasoning approaches have been defined, that

G. Serra and C. Tasso (Eds.): IRCDL 2018, CCIS 806, pp. 57–68, 2018.
https://doi.org/10.1007/978-3-319-73165-0_6

allow to understand which subsets of arguments in the graph are mutually compatible, based on the fact that they are able to defend each other from attacks of other disputing arguments. Some of these approaches may handle weighted graphs/edges.

This paper proposes an approach to extractive text summarization based on abstract argumentation. Attacks represent the fact that two sentences cannot be both included in the summary. *Vice versa*, supports represent the fact that two sentences should be both included in the summary. The set of 'consistent' arguments/sentences computed by argumentation should represent a suitable summary. Attacks and supports are set based on the degree of similarity between two sentences: indeed, different sentences are likely to cover a larger portion of the original text, while similar sentences are likely to bear much redundancy. In this perspective, the weight on the graph edges (i.e., the kind and strength of the relationship between sentences/arguments) might be determined using some similarity measure. In a nutshell, the underlying idea is to place an attack relation between pairs of sentences whose similarity is high, in order to enforce the diversity property. *Vice versa*, a support relation is introduced between pairs of sentences with low similarity, in order to enforce the coverage property.

This paper is organized as follows. The next two sections lay out the background of our research. Section 4 introduces our proposals, and Sect. 5 evaluates its performance. Finally, Sect. 6 concludes the paper.

2 Related Work on Text Summarization

Extractive Text Summarization methods are usually performed in three steps [18]: (i) creation of an intermediate representation of the input which captures only the key aspects of the text (by dividing the text into paragraphs, sentences, and tokens); (ii) scoring of the sentences based on that representation; and (iii) generation of a summary consisting of several sentences, selected by appropriate combination of the scores computed in the previous step.

The score of sentences should be computed using a measure that is able to express how significant they are to the understanding of the text as a whole. For instance, [10] proposed to score sentences using a new measure that expresses their similarity. Its computation encompasses three linguistic layers: (i) the lexical layer, which includes lexical analysis, stopwords removal and stemming; (ii) the syntactic layer, which performs syntactic analysis; and (iii) the semantic layer, that mainly describes the annotations that play a semantic role. These three layers handle the two major problems in measuring sentence similarity, i.e., the meaning and word order problems, in order to automatically combine different levels of information in the sentence while assessing similarity.

In this setting, many strategies have been proposed in the literature to determine which sentences in a given text can be considered as representative of its content. *Word scoring* approaches [12] assigns scores to the most important words. On the other hand, *sentence scoring* approaches determine the features of sentences by detecting and by leveraging the presence of cue-phrases [21] and numerical data [9]. Finally, *graph scoring* analyzes the relationships between the sentences that make up the text. The TextRank algorithm [16] extracts important keywords from a text document, where the weight expressing the importance of a word within the entire document is determined using an unweighted graph-based model.

Some efforts spent in combining the various approaches prove that hybrid approaches may lead to better results. Indeed, [9] shows that combining scoring techniques leads to an improvement in the performance of both single- and multi-document summarization tasks, as measured by the traditional metrics used in this setting (ROUGE scores —see next). In general, the same techniques used in single document summarization systems are applicable to multi-document ones.

Finally, in order to form a paragraph length summary, one approach is based on *Maximal Marginal Relevance* [2], in which the best combination of important sentences is selected.

3 Abstract Argumentation

Since the approach to extractive text summarization that we will propose in this paper is based on argumentative reasoning, we first recall here the basics of this inference strategy. As said, argumentation is an inferential strategy that aims at selecting reliable items in a set of conflicting claims (the *arguments*). In particular, we will consider Abstract Argumentation, which neglects the actual content and inner structure of arguments, to focus just on their external relationships of attack (or, sometimes, support). These relationships may be expressed by a graph in which nodes represent arguments, and edges represent attack or support relationships among arguments.

One of the most influential computational models of arguments proposed in this setting is represented by Dung's Argumentation Frameworks [6] (AFs for short), defined as follows.

Definition 1. *An* Argumentation Framework *(**AF**) is a pair* $F = \langle \mathcal{A}, \mathcal{R} \rangle$*, where* $\mathcal{A}$ *is a finite set of arguments and* $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$*. The relation* $a\mathcal{R}b$ *means that* a *attacks* b*.*

There are a few central concepts when evaluating the justification of an argument:

Definition 2. *Let* $F = \langle \mathcal{A}, \mathcal{R} \rangle$ *be an AF and* $S \subseteq \mathcal{A}$*. Then:*

- S *is* conflict-free *if* $\nexists a, b \in S$ *s.t.* $a\mathcal{R}b$*;*
- $a \in \mathcal{A}$ *is* defended by S *if* $\forall b \in \mathcal{A}\colon b\mathcal{R}a \Rightarrow \exists c \in S$ *s.t.* $c\mathcal{R}b$*;*
- $f_F\colon 2^{\mathcal{A}} \mapsto 2^{\mathcal{A}}$ *s.t.* $f_F(S) = \{a \mid a$ *is defended by* $S\}$ *is called the* characteristic function *of* F*.*
- S *is* admissible *if* S *is conflict-free and* S *is defended by itself, i.e.* $\forall a \in S, \forall b \in \mathcal{A} : b\mathcal{R}a \Rightarrow \exists c \in S$ *s.t.* $c\mathcal{R}b$*.*

The conditions for arguments acceptance are defined by different semantics. Semantics produce acceptable subsets of the arguments, called *extensions*, that correspond to various positions one may take based on the available arguments. Standard acceptability semantics characterize admissible sets of arguments:

Definition 3. *Let $F = \langle \mathcal{A}, \mathcal{R} \rangle$ be an AF and $S \subseteq \mathcal{A}$ be an admissible set. Then, S is a:*

- complete *extension iff* $S = f_F(S)$*;*
- grounded *extension iff S is the $\subseteq$-minimal complete extension.*
- preferred *extension iff S is a $\subseteq$-maximal complete extension.*
- stable *extension iff* $\forall a \in \mathcal{A}, a \notin S, \exists b \in S$ *s.t.* $b\mathcal{R}a$*.*

The justification state of an argument can be conceived in terms of its extension membership. A basic classification encompasses only two possible states for an argument, namely justified or not justified. In this respect, two alternative types of justification, i.e. skeptical or credulous, can be considered.

Definition 4 (Justification State). *Let $F = \langle \mathcal{A}, \mathcal{R} \rangle$ be an AF, and $\mathcal{E}_\sigma(F) = \{S \subseteq \mathcal{A} \mid \sigma(S)\}$ be the set of extensions for a given semantics σ ($\sigma \in \{\text{complete}, \text{grounded}, \text{preferred}, \text{stable}\}$). Then, an argument $a \in \mathcal{A}$ is:*

- skeptically justified *iff* $\forall E \in \mathcal{E}_\sigma(F) : a \in E$*;*
- credulously justified *iff* $\exists E \in \mathcal{E}_\sigma(F) : a \in E$*.*

Many strategies can be found in the literature for the identification of the successful arguments in an argumentation dispute [3,7,19].

A Bipolar AF (*BAF*) [3] is an extension of Dung's AF in which two kinds of interactions between arguments are possible: attack and support. These two relations are independent and lead to a bipolar representation of the interaction between arguments. A BAF can be represented by a directed graph in which two kinds of edges are used, in order to differentiate between the two relations.

Definition 5. *A* BAF *is a triplet $B = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$, where $\mathcal{A}$ is a set of arguments, $\mathcal{R}_{att}$ is a binary relation on $\mathcal{A}$ called attack relation and $\mathcal{R}_{sup}$ is another binary relation on $\mathcal{A}$ called support relation. For two arguments a and b, $a\mathcal{R}_{att}b$ (resp., $a\mathcal{R}_{sup}b$) means that a attacks b (resp., a supports b).*

In BAFs, new kinds of attack emerge from the interaction between the direct attacks and the supports: there is a *supported attack* for an argument b by an argument a iff there is a sequence of supports followed by one attack, while there is an *indirect attack* for an argument b by an argument a iff there is an attack followed by a sequence of supports. In particular, we say that a supports b if there is a sequence of direct supports from a to b. Taking into account sequences of supports and attacks leads to the following definitions applying to sets of arguments [3].

Definition 6. *Let $B = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ be a BAF. A set $S \subseteq \mathcal{A}$* set-attacks *an argument $b \in \mathcal{A}$, iff there exists a supported attack or an indirect attack for b from an element of S.*

A set $S \subseteq \mathcal{A}$ set-supports *an argument $b \in \mathcal{A}$, iff there exists a sequence $a_1 \mathcal{R}_{sup} \dots \mathcal{R}_{sup} a_n$, $n \geq 2$, such that $a_n = b$ and $a_1 \in S$.*

A set $S \subseteq \mathcal{A}$ defends *an argument $a \in \mathcal{A}$, iff for each argument $b \in \mathcal{A}$, if $\{b\}$ set-attacks a, then b is set-attacked by S.*

In the following, we define the semantics for acceptability in BAFs.

Definition 7. *Let $B = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ be a BAF and $S \subseteq \mathcal{A}$. Then, S is:*

- conflict-free*, iff $\nexists a, b \in S$ s.t. $\{a\}$ set-attacks b;*
- safe*, iff $\nexists b \in \mathcal{A}$ s.t. S set-attacks b and either S set-supports b or $a \in S$;*
- *a* d-admissible *extension, iff S is conflict-free and $\forall a \in S, a$ is defended by S;*
- *an* s-admissible *extension, iff S is safe and $\forall a \in S$, a is defended by S;*
- *a* d-preferred *(resp.* s-preferred*) extension is a $\subseteq$-maximal d-admissible (resp. s-admissible) subset of $\mathcal{A}$.*

A weighted AF (WAF) [7] is another extension of Dung's AF in which attacks between arguments are associated with a weight, indicating the relative strength of the attack. In this framework, some inconsistencies are tolerated in subsets S of arguments, provided that the sum of the weights of attacks between arguments of S does not exceed a given inconsistency budget $\beta \in \mathbb{R}^+_*$. The meaning is that attacks up to a total weight of β are neglected. Dung's argument systems assume an inconsistency budget of 0, while, by relaxing this constraint, WAFs can achieve more solutions.

4 Argumentation-Based Text Summarization

Our summarization framework consists of the following phases:

Natural Language pre-processing. The document d to be summarized is progressively splits into sentences $\langle s_1, s_2, \dots, s_n \rangle$, then each sentence s_i is split into a sequence of tokens (words) $\langle s_{i,1}, s_{i,2}, \dots, s_{i,k} \rangle$, and finally each token undergoes lemmatization and stopword removal.

Weighted graph building. The similarity between each pair of sentences is computed and exploited to generate a weighted graph $G = (V, E, f_w)$ where nodes $V = \{s_1, s_2, \dots, s_n\}$ are the sentences in d, and edges $E \subseteq V \times V$ are weighted by the degree of similarity between the associated sentences as computed by the weighting function $f_w \colon E \to \mathbb{R}$.

Argumentation Framework building and evaluation. The resulting graph G is used to build an argumentation framework F_G expressing (possibly weighted) attacks and supports, on which computing a semantics that will determine which sentences are to be selected for inclusion in the summary.

Each phase can be implemented using different approaches and techniques. In the following we will focus on the last phase. Designing this phase requires to choose 3 components: (i) the Argumentation Framework setting; (ii) the graph transformation procedure that builds F_G starting from G; and (iii) the semantics for computing the summary.

As to the first issue, in this preliminary investigation we focused on BAFs, as the simplest AF that allows to consider both attacks and supports between arguments. Concerning the second issue, to derive supports and attacks starting from the weighted edges in G, we defined a heuristic, inspired by the concept of *inconsistency budget* in weighted argumentation frameworks [7]. Intuitively, we want to consider as supports arcs connecting sentences which are dissimilar from each other (because, in some sense, they may bear disjoint information that is worth including in the summary in order to enforce the *coverage* property), and to set attacks between pairs of similar sentences (to model the fact that including both in the summary would not bring much additional information to the summary, violating the *diversity* property). So, we normalize to $[0, 1]$ the weights in G and define two thresholds in order to distinguish which edges in G are attacks or supports in F_G:

- the *attack threshold* $\alpha \in [0, 1]$ and
- the *support threshold* $\beta \in [0, 1]$,

with $\beta < \alpha$. Then, we generate $F_G = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ such that $\mathcal{R}_{sup} = \{e \in E \mid f_w(e) \leq \beta\}$, $\mathcal{R}_{att} = \{e \in E \mid f_w(e) \geq \alpha\}$, and $\mathcal{A} = \{v \in V \mid \exists u \in V : (v, u) \in \mathcal{R}_{sup} \cup \mathcal{R}_{att} \vee (u, v) \in \mathcal{R}_{sup} \cup \mathcal{R}_{att}$. So, we place attacks between very similar sentences, supports between very dissimilar ones, and leave an intermediate similarity range for which we do not set attacks nor supports. Finally, for the third issue, once the BAF is instantiated, we considered the following extension-based semantics (listed from the most credulous to the most skeptical ones) to evaluate the acceptability of arguments: *d-admissible*, *s-admissible*, *complete*, *d-preferred*, *s-preferred* and *stable*. What we expect from the arguments evaluation is that:

- *conflict-free* sets collect the largest subsets of arguments encompassing the main principle of argumentation solutions, i.e., the idea that winning arguments should not attack each other. This requirement would reward sentences that maximize the diversity property. However, for the coverage property, we require that a solution defends its element, too.
- *d-/s-admissible* sets collect a large number of arguments. In principle, these solutions may be appropriate for text summarization tasks. However, when the allowed length of the summary is constrained by an upper bound, admissible solutions may include too many arguments, thus yielding a summary which is ideally good but exceeds the allowed length.
- *complete* extensions collect sets of arguments which can defend all and only their elements from external attacks. They are still admissible and will achieve at most as many solutions as the admissible ones. However, complete semantics may include sets of arguments that are too small.

- *d-/s-preferred* extensions collect sets of arguments maximally-included in d-/s-admissible sets. Therefore, preferred semantics should in principle behave better than admissible and complete ones, both in terms of quality of the summary and in terms of summary length.
- *stable* extensions collect sets of arguments that are able to attack all the remaining arguments not included in the set. In terms of summary requirements, they contain the most dissimilar sentences. Due to their strong requirements, stable semantics might not achieve any solution at all.

5 Evaluation

The effectiveness of the proposed approach was evaluated on the *single-document text summarization* task of the English dataset of the *MultiLing 2015* challenge [11]. This allowed us to have both the ground truth and the state-of-the-art results, published after the competition, available for testing and comparison purposes. On average, the input texts were made up of about 25542 characters, while the ground truths were made up of about 1857 characters (i.e., about 7% of the source texts) on average. Following the experimental protocol defined for the challenge, two variants of the ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) [13] measure were used to quantitatively evaluate the generated summaries: *ROUGE-1* and *ROUGE-2*, where ROUGE-N is the N-gram recall between a candidate summary and the reference summary. So, ROUGE-1 and ROUGE-2 consider the ratio of co-occurring unigrams (i.e., single words) and bi-grams (i.e., pairs of adjacent words), respectively, in a candidate summary over the reference summaries.

In this respect, it is important to point out that the ground truth in this dataset is obtained by humans using an abstractive approach. This deserves some discussion. First, the comparison under these conditions is partly unfair, because there is no exact match between the sentences in the input text and those in the summary. Indeed, since the summary authors were free to merge and restructure in one sentence several parts of the input texts, possibly belonging to many different sentences, the chances of obtaining the same results are significantly affected. Even worse, when inspecting the summaries, it turned out that they may include words that were not present at all in the input texts (e.g., full spelling of the acronyms). This means that, using an extractive summarization approach, it might be impossible to match exactly the ground truth summary, even considering the whole input text. Last but not least, condensing the original content into just 7% opens significant possibilities that the authors of the summary have taken many subjective decisions about what to include in, and what to filter out from, the ground truth, leaving the possibility that other summaries might be as good as theirs, but quite different from theirs.

In the following, we compared our method with the following baselines, taken from the published results of the *MultiLing 2015* competition, whose performance is reported in Table 1:

Table 1. Experimental evaluation results for the MultiLing 2015 dataset

	ROUGE-1	ROUGE-2
WORST	37.17%	9.93%
BEST	50.38%	15.10%
ORACLE	61.91%	22.42%

WORST the worst-performing approach of the challenge;
BEST the best-performing approach of the challenge;
ORACLE an upper bound on the extractive text summarization performance: it uses a covering algorithm [5] that selects sentences from the original text covering the words in the summary disregarding the length limit.

These approaches are set so as to return summaries having the exact length in characters as the ground truth. This is questionable as well, since by truncating a candidate summary to a pre-defined number of characters spoils the very aims and motivations of summarization, which is returning a shorter version of the text that still conveys most of the original content and is human-understandable.

On the other hand, as already pointed out, argumentation semantics return subsets ('extensions') of arguments (sentences) that are mutually consistent ('justified'). This means that, using our argumentation-based approach, (i) we have no control on the number of sentences that are selected to make up the summary, except for choosing different semantics that tend to return larger or smaller extensions; and (2) the control is at the level of sentences, whereas in the challenge length comparison to the ground truth is made in terms of characters. For semantics that returned several extensions, in the spirit of summarization, the shortest one was adopted.

Concerning the first two phases of the approach, we adopted the same solutions as in [8], that we will quickly recall in the following for the sake of illustration. The Natural Language pre-processing phase was mainly based on the *Stanford CoreNLP* toolkit [15], including the dependency parser [4] to extract additional information about word dependency. The *Simplified Lesk* algorithm [23] was also used for word sense disambiguation based on *Wikipedia* or *WordNet* [17], and word embeddings were computed. As regards graph building, the weight between two sentences is computed based on the similarity of their building tokens computed using a combination of three different similarity functions: a *syntactic similarity* based on the *Jaccard Index* applied to syntactic dependencies; a *semantic similarity* based on the function proposed in [20] for taxonomic information based on the synsets in *WordNet*, and an *embedding similarity* based on *cosine similarity* between the word embeddings.

As regards the argumentation phase, we built several argumentation frameworks by setting different values for thresholds α and β. Specifically, we considered values for the support threshold β ranging into $[0.1, 0.8]$, and values for the attack threshold α ranging into $[0.15, 0.9]$, using a 0.05 step for both and ensuring that $\beta < \alpha$ in each setting. All valid threshold combinations resulted

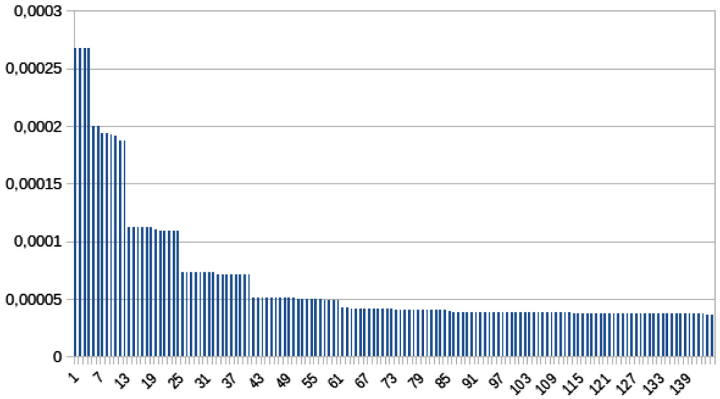

Fig. 1. Quality results for the different argumentation settings on the MultiLing 2015 dataset

in several hundreds of argumentation frameworks for each summarization task, on each of which we computed all the extension-based semantics selected in the previous section: d-/s-admissible, d-/s-preferred, stable and complete. Occasionally, we also computed conflict-free sets, that might provide the most suitable trade-off between acceptability membership conditions and justification state of arguments.

To have an immediate idea of the balance between the summarization performance and the length of a summary s, we defined a compound indicator:

$$\text{Quality}(s) = \text{ROUGE-1}(s)/\text{length}(s)$$

where the length of the summary, placed at the denominator, penalizes the quality of the solution s. Figure 1 graphically summarizes the results: each item on the x axis corresponds to a summarization task, for which the y axis reports the average Quality value obtained on all settings run for that task. Tasks are ordered on the x axis by decreasing average Quality. For the sake of clarity, the graph considers only 144 (σ, α, β) settings, that were selected as representative of the whole behavior. Specifically, $\sigma \in$ {s-admissible, d-admissible, stable, complete} is the semantics, and $\beta \in [0.1, 0.8]$ and $\alpha \in [0.2, 0.9]$, using a 0.1 step, are the thresholds in which the constraint $\beta < \alpha$ is satisfied for the semantics σ.

While in most of the graphic the decay in performance is smooth, it is also apparent that 4 'steps' naturally emerged, associated to sudden drops in quality, as if a phase transition occurred. So, we leveraged these steps to select the most relevant settings to investigate in more depth. Each step corresponds to two different semantics that returned exactly the same results, as reported in Table 2. Note that summaries associated to the first step are shorter than the ground truth (1352 characters against 1857), then in the second step the length of the summaries jumps from 1352 to 4365 characters. So, there was no summary whose length was close to that of the ground truth (1857 characters). The ROUGE-1

Table 2. Experimental evaluation results for our approach. Average size of full texts is 25542 characters, average size of the ground truths is 1857 (7% of the full texts)

Step	Semantics	α	β	Length (%)	Quality	Rouge-1		Rouge-2	
						Recall	Precision	Recall	Precision
1	s-admissible d-admissible	0.1	0.3	1279 (5%)	2.00E-04	25.57%	41.97%		
2	s-admissible d-admissible	0.1	0.4	4365 (17%)	1.13E-04	49.28%	30.32%	15.49%	7.22%
3	stable complete	0.1	0.5	8544 (33%)	7.07E-05	60.44%	24.44%	23.98%	7.43%
4	s-admissible d-admissible	0.1	0.5	9826 (38%)	7.33E-05	72.09%	26.65%	27.26%	6.16%

results are comparable to the state-of-the-art at step 2, comparable to Oracle at step 3, and much (>10%) better than Oracle at step 4, but using the 17%, 33% and 38% of the input texts, respectively (compared to 7% of the ground truth). In other words, the argumentation approach requires more than twice as many characters as the ground truth to obtain the same ROUGE-1 results as the state-of-the-art, and 1/3 of the whole text to reach Oracle. By allowing it to use a little more text, but however less than 2/5 of the input text, it is able to catch nearly 3/4 of the content. Considering ROUGE-2, the same comments as above still hold, but in this case the recall value is slightly larger than the reference systems. Also the results at step 1 are interesting: even if the length of the summary is less than that of the ground truth, recall is not so bad, and precision is quite high. ROUGE-2 was not computed for the first step, due to the summary being very short.

These results suggest that the proposed Argumentation-based approach is sensible and effective in returning relevant summaries, and is competitive in performance, albeit paying in summary length. Confirming our hypothesis, s-preferred and d-preferred semantics provide relevant results. However, also the stable and complete semantics may yield interesting results that somehow represent a trade-off corresponding to the performance of ORACLE.

Given the considerations about possible unfairness of the ground truth construction, we wanted to carry out also a qualitative evaluation of our summaries, by asking human beings to read them and provide their sensations. Very interestingly, they reported that the proposed summaries have little redundancy, yet provide a sensible account of the original document, also ensuring smooth discourse flow, even if they were obtained by filtering out sentences that, since present in the original text, presumably included relevant parts as regards the content and/or the flow of discourse.

6 Conclusion

The ever-increasing number of text documents that are present in digital libraries makes it impossible for humans to read and understand them in order to assess their relevance and/or grasp the content they express. Automatic text summarization is a possible solution, aimed at using computers to extract automatically summaries of the input text(s) that preserve their fundamental meaning. Thus, providing effective solutions to this task would bring enormous benefit to the library users. This paper focused on extractive text summarization, aimed at selecting subsets of sentences taken from the original documents that are necessary and sufficient to cover their content. Specifically, it proposed a framework whose core step is carried out using abstract argumentation, based on a similarity assessment between pairs of sentences in the document.

Experimental results obtained on the English subset of the benchmark MultiLing 2015 dataset confirmed the viability and effectiveness of the proposed approach. Differently from other approaches in the literature, the argumentation-based approach autonomously determines the number of sentences to be included in the summary, which is typically larger than required by the dataset's ground truth. However, the summaries are still significantly shorter than the original text, and reach very high performance. Being the approach general, we expect that similar results can be obtained on other languages, as well.

Future work will carry out further investigations on the possibility of improving the performance of the approach by exploring other argumentation frameworks (e.g., those that may handle weights on attacks and supports) and semantics. Also, further experiments on additional datasets and languages are planned.

Acknowledgments. This work was partially funded by the Italian PON 2007–2013 project PON02_00563_3489339 'Puglia@Service'.

References

1. Banerjee, S., Mitra, P., Sugiyama, K.: Multi-document abstractive summarization using ILP based multi-sentence compression. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence, IJCAI 2015, pp. 1208–1214 (2015)
2. Carbonell, J., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: ACM SIGIR, pp. 335–336. ACM (1998)
3. Cayrol, C., Lagasquie-Schiex, M.C.: On the acceptability of arguments in bipolar argumentation frameworks. In: Godo, L. (ed.) ECSQARU 2005. LNCS (LNAI), vol. 3571, pp. 378–389. Springer, Heidelberg (2005). https://doi.org/10.1007/11518655_33
4. Chen, D., Manning, C.D.: A fast and accurate dependency parser using neural networks. In: EMNLP, pp. 740–750 (2014)
5. Davis, S.T., et al.: OCCAMS-an optimal combinatorial covering algorithm for multi-document summarization. In: ICDMW, pp. 454–463. IEEE (2012)

6. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**(2), 321–357 (1995)
7. Dunne, P.E., et al.: Weighted argument systems: Basic definitions, algorithms, and complexity results. Artif. Intell. **175**(2), 457–486 (2011)
8. Ferilli, S., Pazienza, A., Angelastro, S., Suglia, A.: A similarity-based abstract argumentation approach to extractive text summarization. In: Esposito, F., Basili, R., Ferilli, S., Lisi, F. (eds.) AI*IA 2017. LNCS, vol. 10640, pp. 87–100. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70169-1_7
9. Ferreira, R., et al.: Assessing sentence scoring techniques for extractive text summarization. Expert Syst. Appl. **40**(14), 5755–5764 (2013)
10. Ferreira, R., et al.: A new sentence similarity assessment measure based on a three-layer sentence representation. In: DocEng, pp. 25–34. ACM (2014)
11. Giannakopoulos, G., et al.: Multiling 2015: multilingual summarization of single and multi-documents, on-line fora, and call-center conversations. In: SIGDIAL, pp. 270–274 (2015)
12. Gupta, P., et al.: Summarizing text by ranking text units according to shallow linguistic features. In: ICACT, pp. 1620–1625. IEEE (2011)
13. Lin, C.: Rouge: A package for automatic evaluation of summaries. In: ACL 2004 Workshop, vol. 8 (2004)
14. Lloret, E., Palomar, M.: Text summarisation in progress: a literature review. Artif. Intell. Rev. **37**(1), 1–41 (2012)
15. Manning, C.D., et al.: The stanford CoreNLP natural language processing toolkit. In: ACL (System Demonstrations), pp. 55–60 (2014)
16. Mihalcea, R., Tarau, P.: Textrank: Bringing order into texts. Association for Computational Linguistics (2004)
17. Miller, G.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
18. Nenkova, A., McKeown, K.: A survey of text summarization techniques. In: Aggarwal, C., Zhai, C. (eds.) Mining Text Data, pp. 43–76. Springer, Heidelberg (2012). https://doi.org/10.1007/978-1-4614-3223-4_3
19. Pazienza, A., Esposito, F., Ferilli, S.: An authority degree-based evaluation strategy for abstract argumentation frameworks. In: Proceedings of the 30th Italian Conference on Computational Logic, pp. 181–196 (2015)
20. Rotella, F., Leuzzi, F., Ferilli, S.: Learning and exploiting concept networks with conNeKTion. Appl. Intell. **42**(1), 87–111 (2015)
21. Shardan, R., Kulkarni, U.: Implementation and evaluation of evolutionary connectionist approaches to automated text summarization. J. Comput. Sci. **6**, 1366–1376 (2010)
22. Umam, K., et al.: Coverage, diversity, and coherence optimization for multi-document summarization. Jurnal Ilmu Komputer dan Informasi **8**(1), 1–10 (2015)
23. Vasilescu, F., Langlais, P., Lapalme, G.: Evaluating variants of the lesk approach for disambiguating words. In: LREC (2004)