Aspect-Based Argument Mining

Dietrich Trautmann
Center for Information and Language Processing
Ludwig Maximilian University of Munich, Germany
dietrich@trautmann.me

Abstract

Computational Argumentation in general and Argument Mining in particular are important research fields. In previous works, many of the challenges to automatically extract and to some degree reason over natural language arguments were addressed. The tools to extract argument units are increasingly available and further open problems can be addressed. In this work, we are presenting the task of Aspect-Based Argument Mining (ABAM), with the essential subtasks of Aspect Term Extraction (ATE) and Nested Segmentation (NS). At the first instance, we create and release an annotated corpus with aspect information on the token-level. We consider aspects as the main point(s) argument units are addressing. This information is important for further downstream tasks such as argument ranking, argument summarization and generation, as well as the search for counter-arguments on the aspect-level. We present several experiments using state-of-the-art supervised architectures and demonstrate their performance for both of the subtasks. The annotated benchmark is available at https://github.com/trtm/ABAM.

1 Introduction

The field of computational argumentation (Slonim et al., 2016) gained a lot of interest in the last couple of years. This is noticeable from both the number of the submitted publications related to this field and also from the high volume of emerging datasets (Aharoni et al., 2014; Levy et al., 2017; Habernal et al., 2018; Stab et al., 2018; Trautmann et al., 2020a), specific task formulations (Wachsmuth et al., 2017; Al-Khatib et al., 2020) and models (Kuribayashi et al., 2019; Chakrabarty et al., 2019).

Similar to aspect-based sentiment analysis (Pontiki et al., 2014), we also see the possibility of breaking down arguments into smaller attributes or meaningful components in the argument mining domain. We consider these components as *aspects* of the arguments. Previous works already utilized aspect-information for several subtasks within the argument mining domain (Fujii and Ishikawa, 2006; Misra et al., 2015; Gemechu and Reed, 2019). However, these works vary significantly in the definition of aspects and do not focus on the aspect-based argument mining explicitly, e.g., employ aspects as a source of side or additional information.

For instance, Fujii and Ishikawa (2006) are mainly focusing on the summarization of opinions, visualizing pro and contra arguments for a given topic. Thereby, the authors are extracting aspects, calling them *points at issue*, and ranking the arguments according to them. However, their approach relies on rule-based extraction solely. In Misra et al. (2015), the authors are proposing summarization methods to recognize specific arguments and counter-arguments in social media texts, to further group them across discussions into *facets* (i.e., aspects) on which that issue is argued. Still, this work is limited to a couple of topics and samples. Finally, Gemechu and Reed (2019) also mention aspects as part of four functional components, where the authors interchangeably label aspects and concepts for the specific words. However, to the best of our knowledge, the authors did not publish their labeled data, making a comparative evaluation of aspect extraction methods impossible. We, in contrast, specifically address the aspect term

Proceedings of the 7th Workshop on Argument Mining, pages 41–52
Barcelona, Spain (Online), December 13, 2020.

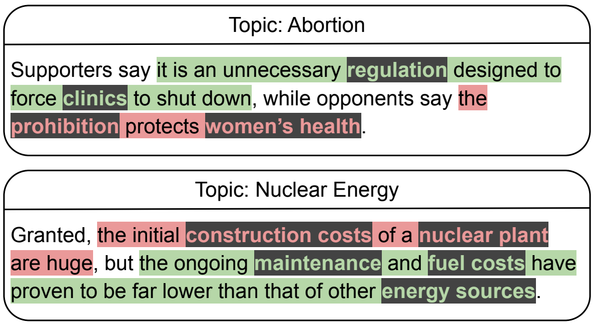

Figure 1: Example annotation of argumentative spans, the corresponding stances (green: supporting/pro; red: opposing/contra) and the aspects (underlined) for the topics *abortion* and *nuclear energy*.

extraction, concentrate on the proper definition of aspects and therefore directly emphasize and present the task of Aspect-Based Argument Mining (ABAM) in this work.

One of the potential applications for the ABAM is the ability to search for specific subtopics within a larger controversial area. For instance, for the topic *abortion*, one can particularly be interested in *regulation* or *health*-related aspects (first example in Figure 1). Whereas for the topic of *nuclear energy*, one can care for solely *enviromental*, *cost*- or *safety*-related aspects (second example in Figure 1). By searching or filtering for the particular aspects, one has the possibility to select for specific information and, therefore, to get more fine-grained results. Another benefit is the ability to compare opposing arguments on the aspect-level.

In this regard, necessary subtasks within the ABAM include the explicit *Aspect Term Extraction* (ATE) on token-level and the *Nested Segmentation* (NS) of argumentative parts along with their aspects within a given sentence. Our work is based on Trautmann et al. (2020a), where the authors already addressed the task of argument unit segmentation. We extend their benchmark with aspect term extraction on these argument units. The ABAM task can be performed in two ways: first, as a two-step pipeline approach with argument unit recognition and classification (AURC) followed by aspect term extraction, or as an end-to-end approach in the form of the nested segmentation task. Since the argument units are already provided by Trautmann et al. (2020a), we can use them directly for the second step in the pipeline, namely the ATE task. Whereas in the end-to-end scenario we adress both tasks (i.e., AURC and ATE) simultaneously for argumentative sentences.

One of the main challenges we faced during this work was the absence of publicly available benchmarks containing the aspect terms. Existing argument mining datasets do not contain the required information and therefore could not be directly applied for *Aspect-Based* Argument Mining. We address this challenge by extending an existing fine-grained argument corpus (Trautmann et al., 2020a) with crowd-sourced token-level aspect information. This is our focused main contribution. While annotating the corpus, we were faced multiple difficulties, including the proper definition of aspects and the creation of rules required for the aspect extraction. It is important to note, that within this work, we refer to aspects as the main point(s) arguments are addressing.

Last but not least, since we are extending the existing corpus, we do not explicitly concentrate on the stance definition and its annotation. Furthermore, as stated in Trautmann et al. (2020a), there are two main argument mining directions: *closed domain discourse-level* and the argument mining from the *information seeking* perspective. The authors of the underlying corpora follow the latter and provide the reasons for that in their work. We, therefore, adopt their vision on that point.

Summarizing the abovementioned points, our contribution within this work is as follows:

- We are emphasizing and presenting the task of Aspect-Based Argument Mining on its own.
- We are extending an existing corpus with token-level aspect terms, making a comparative evaluation of ABAM methods possible.
- We are presenting a number of strong baselines with a corresponding error analysis.

2 Problem Statement

We define the ABAM task as following: Given a list of several topic related texts (documents or paragraphs), we segment the texts into N sentences

$$sentence_i = [t_1, t_2, t_3, \ldots, t_n] \tag{1}$$

The problem is to select, if available, one (or several) span(s)

$$span_j = [t_k, \ldots, t_l] \tag{2}$$

inside each $sentence_i$, with $k >= 1$, $l <= n$, $l - k >= SEG_{min}$ and $l - k <= SEG_{max}$ (with $SEG_{min} = 3$ tokens and $SEG_{max} = n$ tokens in a segment), and a corresponding stance

$$stance_j \in [PRO, CON] \tag{3}$$

Tokens outside of argumentative spans are assigned the NON stance label. Furthermore, regularly there is at least one aspect in every selected span with

$$aspect_j = [t_p, \ldots, t_q] \tag{4}$$

where $p >= k$, $q <= l$, $q - p >= ASP_{min}$ and $q - p <= ASP_{max}$ (with $ASP_{min} = 1$ token and $ASP_{max} = 5$ tokens per aspect).

3 Related Work

Regarding the abovementioned problem definition (§2), we selected three research areas as thematically closed to our task.

Sentiment Analysis: The SemEval workshop organized the task of aspect-based sentiment analysis (Pontiki et al., 2014; Pontiki et al., 2015; Pontiki et al., 2016). Its subtasks also involved the aspect term extraction, which mainly inspired our approach and definition of the aspect term. Recent works applied adversarial training of pretrained language models (Karimi et al., 2020) and a combination of contextualized embeddings and hierarchical attention (Trusca et al., 2020) for new state-of-the-art results on this tasks.

Argument Mining: In our work we adopt the definition of argument facets from the previous work and adjust it for our task. For instance, Misra et al. (2015) used the information on argument facets for the summarization of arguments in social media. Furthermore, the authors used argument facets for the argument similarity task (Misra et al., 2016). The abovementioned works were a first approach in the area of argument facet extraction and were limited to solely a couple of topics and samples. Recent work extended this approach to 28 topics and used the aspect information for the argument similarity task and argument clustering (Reimers et al., 2019). However, the focus of Reimers et al. (2019) was on the pairwise classification of argumentative sentences and not on the aspect term extraction task itself. Lastly, the work by Bar-Haim et al. (2020) defined argument key-points to create concise summaries from a large set of arguments.

Nested Named Entity Recognition: The task of nested-NER is similar to the nested segmentation task (§5.1.2) that we propose. Early work (Finkel and Manning, 2009) presented newspaper and biomedical corpora, and modeled the data by manual feature extraction. Recent works proposed recurrent neural networks (Katiyar and Cardie, 2018) and sequence-to-sequence (Straková et al., 2019) approaches. The latter modeled nested labels as multilabels, a method that we also adopted for our task with overlapping stance and aspect labels.

NN	NNS JJ NNS	JJ HYPH NN NN
NNS	NN POS NN	JJ HYPH NN NNS
NN NN	NN POS NNS	JJ HYPH JJ NN
NN NNS	NNS POS NN	JJ HYPH JJ NNS
JJ NN	NNS POS NNS	JJ JJ NN NN
JJ NNS	IN NN NN	JJ JJ NN NNS
NN NN NN	IN NN NNS	JJ NN HYPH NN
NN NN NNS	JJ NN NN	JJ NN HYPH NNS
NN IN NN	JJ NN NNS	JJ NN JJ NN
NN IN NNS	JJ JJ NN	JJ NN JJ NNS
NN HYPH NN	JJ JJ NNS	JJ NN NN NN
NN HYPH NNS	NN HYPH NN NN	JJ NN NN NNS
NN JJ NN	NN HYPH NN NNS	JJ HYPH NN NN NN
NN JJ NNS	NN POS JJ NN	JJ HYPH NN NN NNS
NNS JJ NN	NN POS JJ NNS	

Table 1: The final set of the 44 Part-of-Speech patterns.

4 Corpus Creation

The creation of the ABAM benchmark is based on the argument units from the AURC corpus (Trautmann et al., 2020a) and is divided into two main parts. The first part addresses two studies for the annotation task formulation, whereas the second part describes the final corpus creation. We outsourced the data annotation to independent (crowd-)annotators and based on their results we created the gold labels.

4.1 Expert Study

We conducted two expert studies on random samples of ten argument units per stance and topic, selected from the AURC corpus. The resulting sets contained 160 samples for each study.

4.1.1 Token-Level Annotation

The first expert study task was to select explicit aspect terms from a given argument unit on the token-level. Two graduate domain experts performed the annotation. Experts were free to select every input-token which fits the following task description: *"The aspects are defined as the most important point(s) the argument unit is addressing"*.

After the annotation step, the Inter-Annotator Agreement (IAA) for the 160 samples was computed. We decided for Cohen's κ (Cohen, 1960) as our agreement measure, that resulted in the initial score of 0.538. According to Viera et al. (2005), this score is in the *moderate agreement* range. Furthermore, the primary analysis of the selected aspect terms from both annotators yielded a list of especially *frequent part-of-speech* (PoS) patterns for the selected tokens. To further improve the annotation process, the PoS information was employed in the second expert study.

4.1.2 Candidates Selection

The aspect candidate selection step is crucial for the correct aspect term extraction task. To select the aspect candidates for the second study, we rely on the part-of-speech information. Specifically, the PoS patterns that occurred more than twice in the previous expert study (i.e., token-level annotation) where picked, and some additional PoS patterns were defined (e.g., the singular and plural form of nouns). The tag set is based on the Part-of-Speech tags used in the Penn Treebank Project[1] and the stanza NLP library[2]. The final PoS pattern list is comprehensive and representative (includes 44 patterns, see Table 1), and ensures linguistically and grammatically correct candidates, without affecting the actual discourse. These PoS patterns were applied on a different set of 160 random samples to create a list of aspect term candidates for every argument unit.

[1] https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
[2] https://stanfordnlp.github.io/stanza/

#	topic	#sentences	#segments	#aspects (total)	#aspects (unique)
T1	abortion	415	435	910	484
T2	cloning	343	365	843	492
T3	marijuana legalization	626	676	1889	887
T4	minimum wage	624	689	1981	745
T5	nuclear energy	615	671	1992	980
T6	death penalty	588	637	1325	545
T7	gun control	480	519	1081	429
T8	school uniforms	705	800	2019	923
	total	4396	4792	12040	**4525**†

Table 2: Count of sentences, segments and (total & unique) aspects in the ABAM corpus. †Clarification: The total count of unique aspects for all topics is 4525, but the sum of all unique aspects per topic is 5485. This is due to some aspects appearing in several topics (c.f. Table 3).

The annotators were asked to solve the same task as before, but now by selecting one or several options from the aspect term candidates list. If none of the aspect term candidates were appropriate, the option *NONE* was selected. This simplification of the task, compared to the first study, led to a raised Cohen's κ of 0.790. This is considered as a *substantial agreement* (Viera et al., 2005) and we deem this as a viable approach for the aspect term extraction.

4.2 Corpus Annotation

Based on the insights from the first two studies, the annotation guidelines (§A) were extended with clearer task formulations and examples. Additionally, the final set of PoS patterns (full list in Table 1) was applied on all argument units from the AURC corpus. The AURC corpus was slightly preprocessed to account for duplicates on the sentence- and segment-level, as well as on some minor errors on span boundaries.

Two independent (crowd-)annotators with a linguistic background and a minimum professional working proficiency in English were recruited for the aspect term extraction task. The annotation procedure was the same as described in §4.1.2. The inter-annotator agreement score for the two expert annotators resulted in a Cohen's κ of 0.874 for all eight (8) topics. This is considered as an *almost perfect agreement* (Viera et al., 2005).

Annotation Merge For the gold standard we selected the annotations where both of the annotators agreed on the token-level. This ensured that we always had a selection of aspects if neither of the annotators selected the NONE option. Additionally, shorter aspect terms are favoured by this annotation merge.

Gold Standard The final descriptive statistics of the ABAM corpus are depicted in the Table 2. There are 12040 aspects in total and 4525 unique (lemmatized) aspects. The topic with the most segments (T8 in Table 2), also yielded the most total aspects (2019). Furthermore, there are 58.10% of the aspects with only one token, 32.12% with 2 tokens, 7.94% with 3 tokens, 1.73% with 4 tokens and only 0.12% with 5 tokens.

Common Aspects In further aspect analysis we aggregated the most common aspects for the eight topics. The top five aspects and the absolute occurence counts per topic, are shown in Table 3. Furthermore, three aspects (*life, problem, government*) appeared in all eight topics and the aspects *people, cost, society, risk, law* appeared in seven topics.

5 Experimental Setup

This section presents our experimental setup regarding the two tasks, the employed models and the data set splits.

topic	aspect (occurrences)
abortion	child (28), life (26), woman (24), unsafe abortion (22), death (16)
cloning	animal (24), child (20), clone (20), disease (16), scientist (16)
marijuana legalization	drug (51), marijuana (44), people (41), alcohol (37), medical marijuana (27)
minimum wage	worker (119), job (52), increase (46), employer (41), economy (39)
nuclear energy	energy (42), electricity (35), fossil fuel (34), environment (30), nuclear power plant (24)
death penalty	crime (62), deterrent (30), punishment (28), cost (27), criminal (27)
gun control	crime (56), gun (55), criminal (28), crime rate (25), gun control law (22)
school uniforms	student (140), parent (77), child (66), kid (60), school (57)
common aspects (in 8 of 8 topics)	life (91), problem (57), government (55)
common aspects (in 7 of 8 topics)	people (94), cost (78), society (51), risk (48), law (42)

Table 3: The top 5 most common aspects per topic and for aspects that appear in several topics.

5.1 Tasks

In this work we apply the two different, but related, sub-tasks for ABAM in the sequence labeling formulation, following Akhundov et al. (2018).

5.1.1 Aspect Term Extraction

In the first task (ATE), we employ only the aspect term information within the segments (argument untis). This sequence labeling task is a *binary* classification problem per token.

5.1.2 Nested Segmentation

In the second task (NS), we utilize full argumentative sentences (like the examples in Figure 1) with the *stance* (PRO, CON, NON) and *aspect* (O, ASP) information for every token as our input. We extend the stance labels with the aspect information for a total set of five possible combinations ([NON,O], [PRO,O], [PRO,ASP], [CON,O], [CON,ASP]).[3]

This is a *multiclass* sequence labeling problem, which solves both the argument unit segmentation and the aspect term extraction tasks.

5.2 Models

BERT For the two subtasks, we decided for the BERT model (Devlin et al., 2019) as a recent state-of-the-art system on a number of natural language processing tasks. We utilize the base and large versions of BERT, as well as both versions of the models with an additional CRF-Layer (Sutton et al., 2012) as the final classification layer in the architecture. Further information about hyperparameter search and computing infrastructure are in §6.2, §B and §C.

PoS Patterns Additionally, we applied the PoS-patterns from the aspect candidates creation step we used in §4. For the ATE task we labeled all tokens that match the PoS-patterns and report the results as the lower boundary of our approaches.

5.3 Evaluation

As the evaluation metric, we report the macro-F1 scores[4] for both of our tasks. Further information about accuracy, precision and recall can be found in §D.

[3]Tokens that are not part of argument units (spans) get the stance-label NON in this sequence labeling task and aspects are always within argumentative spans.

[4]`https://github.com/chakki-works/seqeval`

set \domain	INNER	CROSS
train	2447	2264
dev	333	516
test	693	1319

Table 4: Sample counts per set and domain for the aspect term extraction task.

set \domain	INNER	CROSS
train	2268	2097
dev	307	478
test	636	1185

Table 5: Sample counts per set and domain for the nested segmentation task.

5.4 Inner-Topic & Cross-Topic

For a better understanding of the model performance, we followed the two different dataset splits (domains) as they were defined for the AURC corpus (Trautmann et al., 2020a). In the inner-topic split we trained, evaluated and tested our models on the same set of topics (T1-T6, Table 2). In the cross-topic split we trained our model on T1-T5, selected the best hyperparameter from the evaluation on T6 and tested on T7 and T8. Detailed sample counts are shown in Table 4 and Table 5 for each task, domain and set.

6 Results

This section presents the results for our tasks as described in §5.1.

6.1 Tasks

6.1.1 Aspect Term Extraction

The best performing options are the $\text{BERT}_{\text{LARGE}}$ models (Table 6). Both of them perform similar, but the one with the CRF-layer is slightly better on the development set for inner-topic and the test set for the cross-topic. The inner-topic scores are higher compared to the more challenging cross-topic set-up, were we evaluate the models on unseen topics. All the models performed much better than the lower boundary from the PoS-Patterns Matches. However, this scores are still bellow the human performance of 0.895. The human performance on this task is based on the results from the second expert study (§4.1.2)

6.1.2 Nested Segmentation

The results for NS (Table 7), show that the $\text{BERT}_{\text{LARGE}}$ model outperforms the other listed approaches, except for the development set in the inner-topic set-up. Furthermore, the cross-topic set-up is also more challenging for this task, compared to the inner-topic setting.

domain	INNER		CROSS	
model \set	dev	test	dev	test
PoS-Patterns Matches	.600	.610	.518	.640
$\text{BERT}_{\text{BASE}}$	.819	.813	.673	.749
$\text{BERT}_{\text{BASE}}$+CRF	.823	.812	.669	.743
$\text{BERT}_{\text{LARGE}}$	.830	**.821**	**.683**	.754
$\text{BERT}_{\text{LARGE}}$+CRF	**.832**	.818	.681	**.756**
human performance	.895			

Table 6: F1 results on the dev and test sets for the inner-topic (INNER) and cross-topic (CROSS) set-ups for the aspect term extraction task.

domain	INNER		CROSS	
model \set	dev	test	dev	test
$\text{BERT}_{\text{BASE}}$	.507	.465	.278	.338
$\text{BERT}_{\text{BASE}}$+CRF	.521	.480	.270	.332
$\text{BERT}_{\text{LARGE}}$	.557	**.520**	**.315**	**.369**
$\text{BERT}_{\text{LARGE}}$+CRF	**.563**	.517	.293	.358

Table 7: F1 results on the dev and test sets for the inner-topic (INNER) and cross-topic (CROSS) set-ups for the nested segmentation task.

6.2 Hyperparameters

For our experimental setup with BERT, we fine-tuned the whole (standard) base and large models, as well as both models with an additional final CRF-Layer. We selected the hyperparameters on the development sets and in particular the learning rate (range: 0.00001 - 0.00009 in 0.00001 steps) and the dropout rate (range: 0 - 0.5 in 0.1 steps). We used grid search, to cover all possible combinations. The model parameters were optimized with AdamW (Loshchilov and Hutter, 2018). The training batch size was 32. Our reported results are the averages from three runs and one epoch took about 1 minute for the base models and less than 2 minutes for the large models on average. We fine-tuned for 10 epochs in the ATE task and for 20 epochs in the NS task. Detailed numbers of the final hyperparameters for each model and task can be found in the tables in the appendix §B.

7 Error Analysis

Recalling our definition of aspects: They are defined as the main point(s) argument units are addressing. Furthermore, considering our annotation guidelines in §A, the most important point is usually not equal to the given main topic. An overview of the main errors found during the evaluation of the development sets for the best performing models in the inner- and cross-topic set-ups, is given below.

Aspect Term Extraction During the evaluation of ATE results, we observed a number of errors, which we grouped into the following categories:

- The models tend to favour NOUNS in general.
- Topic words, such as *abortion* or *marijuana legalization*, are often selected as aspects, which is in conflict with our guidelines.
- Phrase constructions like *thread of ...* are often selected as a whole aspect by the models. For the benchmark, we, in contrast, focus on the main representative word of such constructions (e.g., *suicide* vs. *thread of suicide*).
- In the case of ADJECTIVE+NOUN, we suggest to avoid general adjectives (e.g. *new* in *new treatments*), whereas focused adjectives that are part of the concept should be selected (e.g. *recreational* in *recreational marijuana*). Our observation is, that models in general could not sufficiently differentiate between such adjectives.
- Models lack the understanding of domain-specific phrasems like *in vitro fertilisation* or *life without parole* and tend to select only the nominalized part of them (e.g., *fertilisation*, *parole*).

Overall the inner-topic set-up achieved much better performace compared to the cross-topic set-up and both models showed significantly better results over the PoS-Patterns Matches baseline. However, in the cross-topic set-up we faced more repeated errors, such as the tendency to select topic words as aspects and not sufficient understanding of domain-specific phrasems.

Nested Segmentation The typology of the main errors in the NS task is similar to the ATE task. Additionally, in the NS task, a number of errors occured due to the wrong assigment of the stance labels, especially in the cross-topic set-up. These results confirm the insight from Trautmann et al. (2020a), where most of the errors arose due to the wrong stance classification. Apparently, the BERT-based models tend to attach to sentiment words for the stance predictions, which is not always correlated.

8 Conclusion

ABAM is a challenging task that, to the best of our knowledge, was not directly addressed before. We made two important contributions: First, we created and released a publicly available benchmark for Aspect-Based Argument Mining. Second, we showcased several baselines for the two subtasks, namely the Aspect Term Extraction and the Nested Segmentation, and performed an elaborative error analysis. We believe that these findings as well as the benchmark are of high potential for further downstream tasks, such as argument ranking, argument summarization and the search for counter-arguments on the aspect-level.

For the future work, we foresee the investigation of unsupervised approaches for the Aspect Term Extraction task, since they showed promising results within the Aspect-Based Sentiment Analysis domain.

Furthermore, it would be of high interest to incorporate topic-specific knowledge (e.g., understanding of phrasems) into the models to address the discussed error types. In another line of work, one could also explore distant supervision (Rakhmetullina et al., 2018) or domain adaptation methods (März et al., 2019), as well as relational approaches (Trautmann et al., 2020b) for this task.

References

Ehud Aharoni, Anatoly Polnarov, Tamar Lavee, Daniel Hershcovich, Ran Levy, Ruty Rinott, Dan Gutfreund, and Noam Slonim. 2014. A benchmark dataset for automatic detection of claims and evidence in the context of controversial topics. In *Proceedings of the First Workshop on Argumentation Mining*, pages 64–68, Baltimore, Maryland, June. Association for Computational Linguistics.

Adnan Akhundov, Dietrich Trautmann, and Georg Groh. 2018. Sequence labeling: A practical approach. *arXiv preprint arXiv:1808.03926.*

Khalid Al-Khatib, Yufang Hou, Henning Wachsmuth, Charles Jochim, Francesca Bonin, and Benno Stein. 2020. End-to-end argumentation knowledge graph construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7367–7374.

Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. 2020. From arguments to key points: Towards automatic argument summarization. *arXiv preprint arXiv:2005.01619.*

Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathy McKeown, and Alyssa Hwang. 2019. AMPERSAND: Argument mining for PERSuAsive oNline discussions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2933–2943, Hong Kong, China, November. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Jenny Rose Finkel and Christopher D. Manning. 2009. Nested named entity recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 141–150, Singapore, August. Association for Computational Linguistics.

Atsushi Fujii and Tetsuya Ishikawa. 2006. A system for summarizing and visualizing arguments in subjective documents: Toward supporting decision making. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 15–22.

Debela Gemechu and Chris Reed. 2019. Decompositional argument mining: A general purpose approach for argument graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 516–526. Association for Computational Linguistics.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. SemEval-2018 task 12: The argument reasoning comprehension task. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 763–772, New Orleans, Louisiana, June. Association for Computational Linguistics.

Akbar Karimi, Leonardo Rossi, Andrea Prati, and Katharina Full. 2020. Adversarial training for aspect-based sentiment analysis with bert. *arXiv preprint arXiv:2001.11316.*

Arzoo Katiyar and Claire Cardie. 2018. Nested named entity recognition revisited. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 861–871, New Orleans, Louisiana, June. Association for Computational Linguistics.

Tatsuki Kuribayashi, Hiroki Ouchi, Naoya Inoue, Paul Reisert, Toshinori Miyoshi, Jun Suzuki, and Kentaro Inui. 2019. An empirical study of span representations in argumentation structure parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4691–4698, Florence, Italy, July. Association for Computational Linguistics.

Ran Levy, Shai Gretz, Benjamin Sznajder, Shay Hummel, Ranit Aharonov, and Noam Slonim. 2017. Unsupervised corpus–wide claim detection. In *Proceedings of the 4th Workshop on Argument Mining*, pages 79–84, Copenhagen, Denmark, September. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Luisa März, Dietrich Trautmann, and Benjamin Roth. 2019. Domain adaptation for part-of-speech tagging of noisy user-generated text. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3415–3420.

Amita Misra, Pranav Anand, Jean E. Fox Tree, and Marilyn Walker. 2015. Using summarization to discover argument facets in online idealogical dialog. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 430–440, Denver, Colorado, May–June. Association for Computational Linguistics.

Amita Misra, Brian Ecker, and Marilyn Walker. 2016. Measuring the similarity of sentential arguments in dialogue. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 276–287, Los Angeles, September. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *10th International Workshop on Semantic Evaluation (SemEval 2016)*.

Aisulu Rakhmetullina, Dietrich Trautmann, and Georg Groh. 2018. Distant supervision for emotion classification task using emoji2emotion. In *Proceedings of the 1st International Workshop on Emoji Understanding and Applications in Social Media (Emoji2018). Stanford, CA, USA. http://ceurws. org*, volume 2130.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 567–578, Florence, Italy, July. Association for Computational Linguistics.

Noam Slonim, Iryna Gurevych, Chris Reed, and Benno Stein. 2016. Nlp approaches to computational argumentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. 2018. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3664–3674. Association for Computational Linguistics.

Jana Straková, Milan Straka, and Jan Hajic. 2019. Neural architectures for nested NER through linearization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5326–5331, Florence, Italy, July. Association for Computational Linguistics.

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

Dietrich Trautmann, Johannes Daxenberger, Christian Stab, Hinrich Schütze, and Iryna Gurevych. 2020a. Fine-grained argument unit recognition and classification. In *The Thirty-Fourth AAAI Conf. on Artificial Intelligence, New York City, NY, USA, AAAI 2020*. AAAI Press, 2.

Dietrich Trautmann, Michael Fromm, Volker Tresp, Thomas Seidl, and Hinrich Schütze. 2020b. Relational and fine-grained argument mining. *Datenbank-Spektrum*.

Maria Mihaela Trusca, Daan Wassenberg, Flavius Frasincar, and Rommert Dekker. 2020. A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. *arXiv preprint arXiv:2004.08673*.

Anthony J Viera, Joanne M Garrett, et al. 2005. Understanding interobserver agreement: the kappa statistic. *Fam med*, 37(5):360–363.

Henning Wachsmuth, Benno Stein, and Yamen Ajjour. 2017. "PageRank" for argument relevance. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1117–1127, Valencia, Spain, April. Association for Computational Linguistics.

A Annotation Guidelines

Annotation guidelines defined for the Aspect Term Extraction task in Aspect-Based Argument Mining.

Task Description

- Given a main topic and an argumentative segment (unit), please select one or several options from the aspect candidates list.
- If no aspect candidate could be selected from the list, pick the option *None*.

While selecting the aspects, please consider the following rules:

- An aspect is defined as the most important/relevant point for the argument made.
- The most important point is usually not equal to the given main topic.
- In case of doubt, shorter aspects candidates (generic terms; e.g. "life span") are prefered over longer candidates (e.g. "prolonged life span").

General Hints

- The selected aspect(s) should be related to the topic in general.
- The presence of AND/OR (usually) denote multiple aspects:
  - If a sentence contains multiple phrases (e.g., "abortion causes breast cancer AND it kills unborn children.");
  - If there is an enumeration and objects connected by AND/OR (e.g. "abortion causes breast cancer, infertility and pain.");
- In the case of ADJECTIVE+NOUN, general adjectives should be avoided (e.g. "new" in "new treatments"), whereas focused adjectives that are part of the concept should be selected (e.g. "recreational" in "recreational marijuana").
- Please, use these test-questions for yourself while annotating:
  - Do you want this argument to be shown to someone, if they select this aspect(s) of the topic, or are other aspect terms in this argument more relevant for the point made?
  - Which words make you understand the argument most?
  - Which words are the most relevant and mainly form the meaning of the argument made?
  - If you would compress the argument into a few most relevant words, which words would that be?

B Hyperparameters

The dropout rate of 0.1 was always the best option. The learning rates for the different models are displayed in Table 8 for the ATE task and in Table 9 for the NS task.

domain	INNER	CROSS
$\text{BERT}_{\text{BASE}}$	$6e-5$	$8e-5$
$\text{BERT}_{\text{BASE}}$+CRF	$9e-5$	$9e-5$
$\text{BERT}_{\text{LARGE}}$	$9e-5$	$9e-5$
$\text{BERT}_{\text{LARGE}}$+CRF	$9e-5$	$8e-5$

Table 8: Hyperparameters (learning rate) for the ATE task.

domain	INNER	CROSS
$\text{BERT}_{\text{BASE}}$	$7e-5$	$5e-5$
$\text{BERT}_{\text{BASE}}$+CRF	$8e-5$	$6e-5$
$\text{BERT}_{\text{LARGE}}$	$5e-5$	$7e-5$
$\text{BERT}_{\text{LARGE}}$+CRF	$7e-5$	$8e-5$

Table 9: Hyperparameters (learning rate) for the NS task.

C Compute Resources

We used Kaggle's Kernels[5] for the processing of the data and Google's Colab[6] for the training (fine-tuning) of our models. The former service offers a single 12GB NVIDIA Tesla K80 GPU, while the latter a single 16GB NVIDIA Tesla P100 GPU.

D Additional Results

The additionally reported numbers for accuracy, precision and recall can be found in the Table 10 for the ATE task, in the Table 11 for the NS task. The numbers are the average from three runs.

domain	INNER						CROSS					
set	dev			test			dev			test		
model \metric	acc.	pre.	rec.	acc.	pre.	rec.	acc.	pre.	rec.	acc.	pre.	rec.
PoS-Patterns Matches	.850	.490	.773	.853	.502	.779	.825	.404	.724	.870	.530	.809
$\text{BERT}_{\text{BASE}}$	.943	.784	.858	.942	.786	.842	.878	.580	.804	.912	.682	.830
$\text{BERT}_{\text{BASE}}$+CRF	.945	.789	.860	.942	.789	.836	.877	.575	.800	.911	.678	.822
$\text{BERT}_{\text{LARGE}}$	.946	.799	.864	.945	.803	.840	.881	.582	.827	.914	.686	.835
$\text{BERT}_{\text{LARGE}}$+CRF	.948	.798	.869	.943	.802	.835	.880	.585	.817	.914	.690	.837

Table 10: Accuracy (acc.), precision (pre.) and recall (rec.) results on the dev and test sets for the inner-topic (INNER) and cross-topic (CROSS) set-ups for the aspect term extraction task. These are the average scores from three runs.

domain	INNER						CROSS					
set	dev			test			dev			test		
model \metric	acc.	pre.	rec.	acc.	pre.	rec.	acc.	pre.	rec.	acc.	pre.	rec.
$\text{BERT}_{\text{BASE}}$	.704	.468	.552	.672	.434	.501	.560	.234	.343	.574	.296	.395
$\text{BERT}_{\text{BASE}}$+CRF	.710	.482	.568	.683	.450	.515	.553	.231	.324	.571	.298	.376
$\text{BERT}_{\text{LARGE}}$	.748	.512	.610	.709	.491	.552	.597	.266	.385	.607	.327	.423
$\text{BERT}_{\text{LARGE}}$+CRF	.749	.524	.608	.702	.491	.547	.575	.248	.358	.594	.320	.407

Table 11: Accuracy (acc.), precision (pre.) and recall (rec.) results on the dev and test sets for the inner-topic (INNER) and cross-topic (CROSS) set-ups for the nested segmentation task (args). These are the average scores from three runs.

[5] `https://www.kaggle.com/kernels`
[6] `https://colab.research.google.com/signup`