End-to-End Argumentation Knowledge Graph Construction

Khalid Al-Khatib,[1] Yufang Hou,[2] Henning Wachsmuth,[3]
Charles Jochim,[2] Francesca Bonin,[2] Benno Stein[1]
[1]Bauhaus-Universitat Weimar, Germany
[2]IBM Research, Ireland
[3]Paderborn University, Germany

Abstract

This paper studies the end-to-end construction of an *argumentation knowledge graph* that is intended to support argument synthesis, argumentative question answering, or fake news detection, among others. The study is motivated by the proven effectiveness of knowledge graphs for interpretable and controllable text generation and exploratory search. Original in our work is that we propose a model of the knowledge encapsulated in arguments. Based on this model, we build a new corpus that comprises about 16k manual annotations of 4740 claims with instances of the model's elements, and we develop an end-to-end framework that automatically identifies all modeled types of instances. The results of experiments show the potential of the framework for building a web-based argumentation graph that is of high quality and large scale.

Introduction

People's lives are packed with situations where they need to form opinions, to shape beliefs, or to make decisions on certain topics. To do so, they typically rely on one of the fundamental types of communication: argumentation (Walton 2010). Lately, developing computational models for argumentation has attracted considerable attention (Stede and Schneider 2018). While existing studies propose approaches for the computational mining (Stab and Gurevych 2014), the assessment (Wachsmuth et al. 2017), and the generation (Hua and Wang 2018) of argumentation, *argumentation knowledge-based approaches* are still scarcely touched — presumably, due to their complexity.

Constructing and employing knowledge graphs has been proven to be effective for many computational tasks related to linguistics and social science. For instance, structured knowledge was utilized as a seed for generating interpretable and controllable texts (Lebret, Grangier, and Auli 2016). Also, knowledge graphs were applied with remarkable success to the identification of fake news (Pan et al. 2018) and to the answering of natural language questions (Huang et al. 2019). Moreover, various studies harvested new knowledge (e.g., common sense) through knowledge graph construction, reasoning, and completion(Sap et al. 2019).

The goal of this paper is to study how to construct and employ an argumentation knowledge graph that is adequate to support computational argumentation tasks such as argument synthesis and argumentative question answering [1]. The underlying process that we propose consists of three high-level steps: (1) modeling the graph and identifying its basic elements, (2) building the graph from texts in accordance with the model in an end-to-end manner, and (3) exploiting the graph for computational argumentation tasks. The paper at hand focuses on the first two steps, while showcasing the potential of the third one.

Inspired by the theory of argumentation schemes (Walton, Reed, and Macagno 2008) as well as by causality research (Guo et al. 2018), we model a graph that captures knowledge about effects between concepts encapsulated in many arguments. A node in the graph represents a *concept instance*, and an edge represents one of two types of relation, either *positive effect* or *negative effect*. We ground each node in a concept from a knowledge base (Wikipedia) and represent its good or bad consequence as an attribute. Figure 1 exemplifies the introduced model elements.

To demonstrate the benefit of the proposed argumentation knowledge graph, consider the following statements:

(a) *Nuclear energy leads to emission decline.*

(b) *Nuclear energy undermines renewable solutions.*

(c) *Renewable solutions tackle climate change and help to decline emission.*

When modeling the knowledge encoded within these statements in our graph, "nuclear energy" has a positive effect on "emission decline" through (a), and a negative effect on "renewable solutions" through (b). The latter in turn has a positive effect on "emission decline" through (c). Now, for a claim such as "nuclear energy is good for emission decline", one could hence derive evidence from the graph to *support* the claim through (a), but also one could *counter-attack* it through (b) and (c). Note that this counter-attack is unlikely to be discovered without modeling the underlying knowledge explicitly as proposed.

[1]Argumentative questions are those which implicitly or explicitly elicit pro/con arguments toward an issue, such as "Why Python is powerful for text mining?"

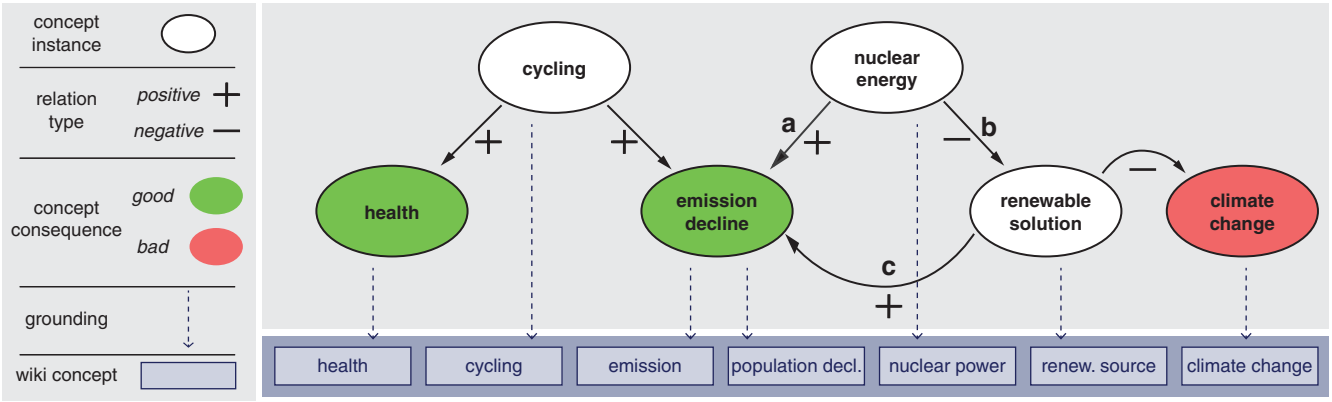

Figure 1: Exemplary instance of the proposed argumentation knowledge graph. Nodes represent concept instances that may have a good or bad consequence, given as an attribute. A directed edge between two nodes indicates a relation in terms of a positive or negative effect of the source on the target concept. Concept instances are grounded in Wikipedia concepts wherever available. The examples a, b, and c in the introduction are marked in the respective edges.

In accordance with our model, we created a new corpus containing 16,429 manual annotations of 4740 selected claims. The annotations include positive and negative effect relations found in the claims, along with their associated concept instances. In addition, we ground the concept instances using Wikipedia, and we explore whether they have a good or a bad consequence in general.

Based on the acquired annotations, we generate two outputs: (1) A new argumentation knowledge graph that consists of 2743 concept instances, 1670 relations, 1705 grounding attributes, and 1513 consequence attributes. (2) A new argumentation corpus for studying the tasks of effect relation detection, relation type classification, and relation concept identification (along with their grounding and consequences). Then, we use this corpus to build a framework for tackling all three tasks end-to-end, i.e., given a set of sentences, we recognize effect-relations along with their types and concepts.

Our approach achieves a macro F_1-score of 0.79 in detecting relations and 0.77 in classifying their types. We apply the framework to a set of sentences belonging to five selected topics from Wikipedia and Annotated English Gigaword (Napoles, Gormley, and Durme 2012). Our inspection of samples of the acquired knowledge shows an average precision of around 0.7. Such results expose the potential of our framework to generate a large-scale web-based argumentation graph.

Overall, the contribution of our paper is three-fold:

- We introduce a new graph model that represents the knowledge encapsulated in arguments.
- We develop a new argumentation corpus, which comprises 16,429 manual labels of 4740 claims.
- We develop a new framework for constructing an argumentation knowledge graph in an end-to-end manner.

The developed resources are freely available on *webis.de*.

Modeling Argumentation Knowledge

Traditional knowledge graphs, such as ConceptNet and Freebase, usually contain known facts and assertions about entities. By contrast, here we are interested in the knowledge specifically needed to support computational argumentation tasks, such as argument generation and reasoning, as sketched in the introductory example above. In particular, we propose to model this knowledge as a directed argumentation knowledge graph with unweighted edges that consists of the following elements:

- *Concept instances* $\sim$ nodes. A concept instance is a phrase expressing an entity ("Donald Trump"), event ("smoking in streets"), or an abstract principle or idea ("society"). If available, concept instances are grounded in concepts from a knowledge base (see below).
- *Effect relations* $\sim$ directed edges. An effect relation is given if some source concept instance affects some target concept, either positively or negatively. The effect relation type (i.e., *positive* or *negative*) is often indicated by cue words in natural text. For instance, "A increases B" indicates that A has a positive effect on B, while "A prevents B" indicates that A has a negative effect on B.
- *Concept consequences* $\sim$ attributes of nodes. A concept instance may be considered, in general, as a *good* or *bad* consequence. Although this is subjective, we expect that there are many concepts for which most people agree on their consequence (i.e., being either good or bad).
- *Concept groundings* $\sim$ attributes of nodes. A concept instance is grounded by mapping it to one or more concepts in a knowledge base. By doing so, concept instances representing the same concept can be identified. For example, the concept instances "capital punishment", "death penalty", and "the death as a penalty of a crime" will all have the grounding of "capital punishment" here.[2]

Figure 1 illustrates all modeled elements of the graph.

[2]https://en.wikipedia.org/wiki/Capital_punishment

Related Work

Several argumentation models have been proposed. One of the well-known theories is *argumentation schemes*. Argumentation schemes are patterns of reasoning that serve as templates for analyzing and creating arguments. The most known scheme set was proposed by Walton, Reed, and Macagno (2008), consisting of 65 different schemes. Our graph model is motivated by two of the five most frequently used schemes (Feng and Hirst 2011), *argument from cause to effect* and *argument from consequences*. While these schemes constitute a solid conceptual foundation for modeling an argumentation graph, their sophisticated structure tends to be hard to match with real-world arguments (Habernal et al. 2018). Instead, our unifying and more straightforward graph representation of positive/negative effects between concepts aims to overcome this shortcoming, in order to achieve broader coverage. For instance, "A has positive effects on B" may imply both causes and consequences, but it can be said without stating a full argument with explicit premises and conclusions. Recently, Lawrence, Visser, and Reed (2019) developed an online annotation assistant to facilitate the annotation of argument schemes. We plan to use this assistant in future work.

For argumentation knowledge graphs, Bex et al. (2013) proposed the argument web, which is a large-scale structure of connected arguments. The structure was implemented using the Argument Interchange Format (AIF), with several integrated tools that form the infrastructure to support argument analysis, exploration, and assessment. In contrast to our work, the AIF tools do not tackle the retrieval of arguments from unstructured sources such as the web.

Toledo-Ronen, Bar-Haim, and Slonim (2016) created a knowledge base of *expert* opinions on debatable topics. This resource, however, connects individuals to debate topics only and cannot be used for relating debate topics or other entities. Saint-Dizier (2016) also argued for using more structured knowledge in argumentation. In his corpus, 78% of the arguments need additional knowledge to link them to the issue discussed. The author states that his generative lexicon qualia structure adequately models this knowledge for argument mining. However, defining qualias is complex and time consuming, so this approach does not scale to the number of arguments we target. Recently, Gemechu and Reed (2019) proposed a new argumentation graph for detecting argument structure that is based on the decomposition of four functional components. In contrast to our graph, the knowledge there concern the premises and conclusions in an argumentative discourse, and the goal is to identify the links between them. Botschen, Sorokin, and Gurevych (2018) used Wikidata as a knowledge base for argument reasoning comprehension, but conclude that world knowledge might not be sufficient and that a more logical analysis is needed. The knowledge graph we propose goes beyond simple world knowledge and has the potential to make reasoning easier.

Outside argument mining, several studies have explored the detection of causality relations (Dunietz, Levin, and Carbonell 2017; Hashimoto et al. 2014; Mirza and Tonelli 2014; Zhao et al. 2017; Dasgupta et al. 2018). Among these, Hashimoto et al. (2014) proposed a method for extracting event causalities from the web using semantic relations. Dunietz, Levin, and Carbonell (2017) tagged causal relations using a hybrid pattern matching and a statistical approach, and Zhao et al. (2017) built a hierarchical causality network to discover high-level abstract causality rules. They embedded the network in a vector space for prediction. Compared to this line of work, besides our concentration on the argumentative context, the effect relations we focus on are more general than the causal relations they target.

The most similar research to our work was done by Reisert et al. (2018) who aim to find simple correspondents of argumentation schemes, specifically arguments from consequences. However, our contribution differs in multiple ways: (1) Our work additionally covers the idea of arguments from cause to effect, (2) our annotation is based on individual sentences while Reisert et al. (2018) focus on pairs of sentences, and (3) only we propose the development of a knowledge graph for computational argumentation tasks.

Manual Knowledge Acquisition

Based on the model presented above, we conducted an annotation study to acquire knowledge for two purposes: (1) constructing an argumentation knowledge graph that is adequate for further completion and extension (Nguyen et al. 2018), and (2) creating a new corpus for the training and evaluation of approaches to the automatic construction of such graphs.

Data Preprocessing and Sampling

In the following, we describe the selected sources and the sampling strategy of the texts in our annotation study.

Data Source We used the complete dataset of Hou and Jochim (2017). The dataset was crawled from the debate portal *Debatepedia* and comprises 25,000 claims along with their (sub)topics, stances, and supporting evidence. *Debatepedia* organizes debate topics hierarchically. For each topic, it contains background information and a number of subtopics, with *pro* and *con* arguments for or against each subtopic. An argument typically includes a boldfaced claim and a few instances of supporting evidence. Our inspection of possible sources in the argumentation space revealed that a large proportion of the claims encode the kind of knowledge we aim to acquire.[3] Each claim forms a self-contained unit, which simplifies the knowledge acquisition process (e.g., it avoids the need for coreference resolution).

Data Sampling To increase the portion of claims that have an effect relation, we compiled a set of potential lexical indicators from the following sources: *+/-EffectWordNet* (Choi and Wiebe 2014), *ConnotationWordNet* (Kang et al. 2014), and *Connotation Frames* (Rashkin, Singh, and Choi 2016). The first two sources provide lists of words and phrases at the sense level and were manually labeled as *positive*, *negative*, and *no effect*. We consider a word as an indicator, if

[3]Note that some claims from *Debatepedia* are more like assertions (e.g., *smoking causes cancer*). We did not filter them out as long as they contain valid positive or negative effect relations.

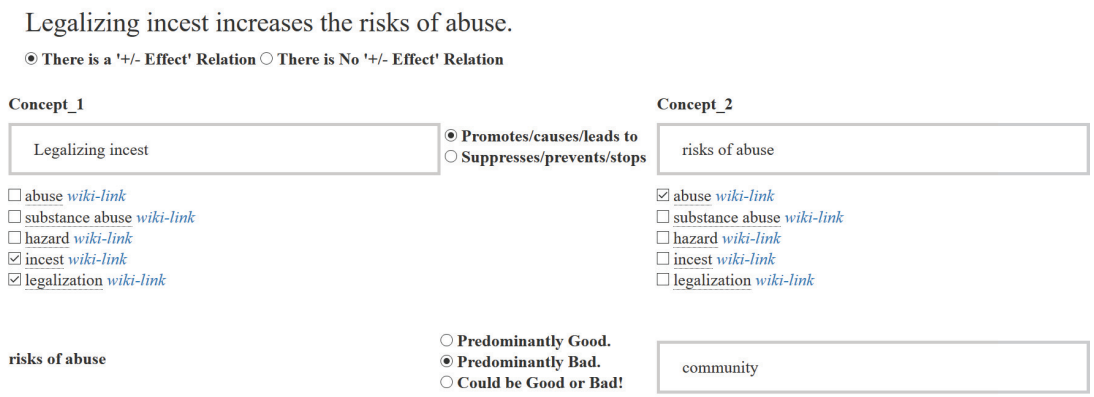

Figure 2: The interface used in the annotation study, along with an example of a worker's annotations.

it has at least one sense that indicates a positive or negative effect. The third source provides a list of words and phrases that are labeled with a set of scores regarding their semantic characteristics. One of these characteristics is the *effect* of the word. We followed the resource developers in mapping the scores to *effect* labels. Each word that is mapped to a positive or negative effect is included in our indicator list.

We extracted all claims with at least one indicator. Then, we excluded those with negated verbs, such as 'don't cause', in order to avoid to mistakenly consider 'no positive effect' as equal to 'negative effect', ending up with 4740 claims. The negation detection was performed using the dependency parser of Dozat, Qi, and D. Manning (2017). To annotate the concepts' grounding, we extracted a candidate list of grounded entities for each selected claim. To this end, we used two wikification tools, TagME (Ferragina and Scaiella 2010) with confidence threshold 0.2 and Babelfy (Moro, Raganato, and Navigli 2014). All entities found by Babelfy were included, as it does not provide confidence scores.

Data Annotation

Using the selected set of claims, we started with a small expert annotation study, followed by several pilot studies. The insights gained from these studies were used to optimize the main annotation study. The annotation task and the different studies are detailed in the following.

Annotation Task The task was to identify the knowledge encapsulated in a given claim, consisting of three core elements: (1) the presence of an effect relation, (2) the concept instances involved in the relation, and (3) the relation type. A concept instance is a phrase that represents an entity (e.g., 'Donald Trump'), an event (e.g., 'smoking in streets'), or an abstract principle or idea (e.g., 'society'). Types of relation between concepts are: (a) positive effect, where a concept instance in the claim promotes/causes/leads to/etc. another concept instance, and (b) negative effect, where a concept instance suppresses/stops/prevents/decreases/etc. another concept instance. To address the grounding of the identified concept instances, we also included to analyze whether the second concept instance in a relation is considered as a 'good' or 'bad' consequence in general. Every annotator was asked to read each given claim thoughtfully and to decide whether it has an effect relation. If a relation was found, the annotator had to mark the two related concept instances in the claim. Then, he/she should select the type of relation.

We provided a set of Wikipedia concepts related to the claim. Each annotator had to examine the concepts either by reading their shown summary or by opening the provided links. A concept should be selected only, if it represented the marked concept instance. If such a concept did not exist, the annotator should select all concepts that, if combined, nearly represent the concept instance. The annotator could decide to not choose any of the concepts, if none of them or their combinations represent the marked concept instance. Next, the annotator had to decide whether the second concept instance is predominantly good, bad, or neutral. In case of good or bad, the annotator should write down some concept instances that the second concept is good/bad for.

The annotators were instructed to pick the primary relation if multiple ones were observed, and to mark the relation if it is manifested explicitly in the claim. For instance, 'governments should prevent crimes' doesn't explicitly manifest a negative effect of 'government' on 'crime'. Figure 2 shows the interface of the annotation task with exemplary annotations. The annotators were provided with examples and were encouraged to leave a comment if they had remarks. To help the workers understand the task, we named the concept instances as *Concept_1* and *Concept_2* in the interface.

Expert Annotation study We asked three argumentation experts to annotate 100 randomly selected claims. After completing the study, the experts confirmed that the task was clear and the interface intuitive. We followed the comments they gave to refine the guideline and to improve the interface. To assess annotation reliability, we computed the inter-annotator agreement in terms of Fleiss' κ and percentage agreement for relation presence and relation type, and in terms of F_1-score for the full and partial overlap (50% of tokens) of the concept instances and their grounded concepts. The obtained agreement ranges from moderate to substantial.

	Relation	Relation Type	Consequence
κ	0.51	0.49	0.45
% Agreement	0.77	0.73	0.69
	Con1	**Con2**	**Entity.Con1**	**Entity.Con2**
Full Overlap	0.58	0.42	0.52	0.46
Par. Overlap	0.67	0.54	0.57	0.52

Table 1: The inter-annotator agreement of the main annotation studies for the annotation labels.

Pilot Crowdsourcing Studies We conducted multiple pilot annotation studies on Amazon Mechanical Turk (AMT). The studies should reveal whether our task suits the crowdsourcing approach, and help to find the best settings for the worker qualifications, the number of annotators per claim, and effective quality control methods. The annotation was done using the same set of claims used in the expert annotation study. We examined various worker requirements, such as acceptance rate, location, and the number of previously accepted hits. Also, we tried different ways to encode check instances. Each claim was annotated by 10 annotators. Besides manually inspecting the annotation quality, we looked at inter-annotator agreement again. In particular, we first aggregated the crowd annotations using majority vote. Then, we computed the agreement between the aggregated crowd annotations and the annotations obtained from the three experts. The obtained agreement here is slightly less than the agreement between experts, but still considered as moderate agreement. The settings with the highest overall agreement were chosen to be used in the main study. For the number of annotators per claim, we considered several subsets of the ten annotations and checked the drop of their agreement compared to the complete set of the ten annotations, if any.

Main Crowdsourcing Study We published five batches of claims for annotation on AMT. Each included about 1000 claims, and every claim was annotated by five workers. Following the observations in the pilot studies, the workers' qualification was set to at least 98% acceptance rate, a minimum of 1000 accepted HITs, and a location in one of the English-speaking countries.

Table 1 shows the inter-annotator agreement of the main study. The agreement is moderate but in line with values reported in comparable tasks (Rashkin et al. 2018). We aggregated relations, their types, as well as the concept instances' consequences using majority vote. Concept instances and grounding concepts were aggregated based on the longest sequence of overlapping words in the annotations.

Annotation Output

The results of the annotation study are a new argumentation knowledge graph and a corpus. We expect both to be valuable resources for the computational argumentation area. The two resources will be made publicly available.

1. Argumentation Knowledge Graph: The entire knowledge acquired was used to construct the graph. The obtained graph includes 2743 nodes and 1670 edges; the construction allows for knowledge completion and expansion.
2. Argumentation Knowledge Corpus: Table 2 shows statistics of the created corpus. About 37% of the claims have an effect relation. The majority of relations is positive, around 74%, and the distribution of good and bad labels similar. The corpus can be used for detecting the sentences with effect relations, as well as for extracting positive and negative effect relations from sentences.

Label	Frequency	Percent
Claim		
Overall	4740	100
Relation	1736	37
No-Relation	3004	63
Relation		
Overall	1736	100
Positive	1287	74
Negative	390	23
No-Agreement	59	03
Consequence		
Overall	1736	100
Good	645	37
Bad	748	43
Neutral	218	13
No-Agreement	125	07
Concept Instance		
Overall	3451	100
Concept 1	1729	50
Concept 2	1722	50
Wikipedia Concept		
Overall	4766	100
Selected Babelfy&Tagme for Con. 1	2229	47
Selected Babelfy&Tagme for Con. 2	2537	53
Labels		
Overall	16429	100

Table 2: Statistics for the argumentation corpus.

Automatic Knowledge Acquisition

This section describes our framework for acquiring the modeled argumentation knowledge *automatically*. The framework targets the tasks of effect-relation detection, relation type classification, consequence classification, and relation concept identification. Using the created corpus, we propose a supervised approach to the tasks of effect-relation detection, relation type classification, and consequence classification. For the task of relation concept[4] identification, we rely on a set of syntactic patterns. In the following, we detail the set of features that we utilize for the tasks at hand.

Relation Detection, Relation Type Classification, and Consequence Classification In essence, *effect* relations are very related to causality relations. The task of causality

[4]For brevity, we simply speak of 'concept' instead of 'concept instance' in this section.

detection has been shown to be challenging, because such a relation can be encoded in diverse text constructions (Dunietz, Levin, and Carbonell 2017). Nonetheless, various types of features proved to be effective there. In our approach, we employ a broad set of features including those which have been applied successfully in causality detection:

1. Lexical Features: Word and character n-grams are powerful features for relation detection and classification as they usually can capture the lexical indicators of relations and their types. In particular, we expect these features to disclose particular verb indicators along with their context. Here, we use *bag of words*, *1–3 word n-grams*, *1–5 char n-grams*, and *tf-idf of word and character n-grams*.
2. Syntax Features: Also, syntax may be important to identify relations. We use the frequencies of *part-of-speech tag 1–3 grams* as well as of *verbs, nouns, adjective, and adverbs*. To obtain part-of-speech tags, we use NLTK.
3. Sentiment Features: We use the *subjectivity of the text* and the *subjectivity of the main verb*. For the first, we use 'Textblob' sentiment analysis. For the second, we use 'SentiWordNet 3.0' (Baccianella, Esuli, and Sebastiani 2010). The hypothesis behind is that the presence of either positive or negative sentiment may be an indicator of the presence of a relation in a claim.
4. Semantic Features: We employ three lexicons that seem useful for the given task: '+/-EffectWordNet' (Choi and Wiebe 2014), 'Connotation Frames' (Rashkin, Singh, and Choi 2016), and 'ConnotationWordNet' (Kang et al. 2014). The lexicons were built manually and extended automatically. Each lexicon comprises words and phrases along with their likely effect (i.e., positive, negative, and null). Recall that these lexicons were used to sample instances in our annotation study. However, they are likely not sufficient for relation identification, as most lexicon entries are verbs, which often have several senses; some may indicate a relation, some not. Nonetheless, two of the lexicons come with effect labels at the sense level. Accounting for that, we employ the following features. *Verb-sense effect* considers the verbs in a text, detects the sense of each verb, and then checks its effect label from the lexicons. Since Word Sense Disambiguation (WSD) is challenging, specifically for verbs (Raganato, Camacho-Collados, and Navigli 2017), we also get the sense distribution of each verb for their effect labels. For example, 'operate' has four senses; three indicate a 'positive' effect, and one a 'null' effect. Based on this distribution, we calculate the *probability of each verb to have a specific effect label*. Also, we compute the *most probable effect of a verb*. The sense of a verb is obtained with multiple WSD methods: the most frequent sense, the knowledge-based WSD 'Lesk' (Banerjee and Pedersen 2002), and the supervised WSD 'IMS' (Zhong and Ng 2010).

Concept Identification For this task, two types of patterns have been used in prior studies: those that are obtained by Open Information Extraction (OpenIE), and those that are derived from Semantic Role Labeling (SRL). In particular, we compose various patterns using the arguments and predicates of OpenIE outputs of Stanovsky et al. (2018) and the SRL outputs of He et al. (2017). For example, one pattern represents the [arg0] of the main verb as the first concept and the concatenation of [arg1] and [arg2] as the second.

	Relation		Type		Consequence	
Feature	**Micr**	**Macr**	**Micr**	**Macr**	**Micr**	**Macr**
Lexical	0.81	0.78	0.86	0.73	0.62	0.45
Syntax	0.73	0.70	0.71	0.57	0.48	0.39
Sentiment	0.66	0.40	0.79	0.44	0.50	0.31
Semantic	0.67	0.56	0.79	0.44	0.60	0.44
All	**0.81**	**0.79**	**0.86**	**0.77**	**0.67**	**0.49**
Baseline	0.66	0.40	0.78	0.44	0.46	0.21

Table 3: The micro and macro F_1-score of our approach in relation detection, relation type classification, and consequence identification, compared to a majority-class baseline.

	Relation		Type		Consequence	
Feature	**Micr**	**Macr**	**Micr**	**Macr**	**Micr**	**Macr**
Lexical	-0.09	-0.09	-0.11	-0.12	-0.06	-0.07
Syntax	0.00	0.00	0.01	0.01	0.01	0.00
Sentiment	0.00	0.00	-0.01	0.00	0.00	-0.01
Semantic	0.01	0.00	0.00	-0.03	-0.02	-0.01

Table 4: Micro and macro F_1-score changes resulting from feature ablation. The results show the impact of removing a feature category compared to using all the features.

Topic	**#Relations**	**Sample-size**	**Precision**
Abortion	5407	60	0.58
Advertising	9213	60	0.73
Global Warming	4927	60	0.67
Social Security	3856	60	0.75
Smoking	4381	60	0.73
All Topics	27784	300	0.69

Table 5: Statistics of the new acquired knowledge.

Experiments and Results

This section presents experiments regarding our framework, along with their results. As for the experimental set-up, we use our argumentation knowledge corpus for training and evaluating the framework. For relation, relation type, and consequence classification, we split the corpus into training (80%) and test (20%) sets where the claims that belong to the same topic exist in either of the two sets. We apply one *support vector machine* with a linear kernel, which performs the best among others we tried, for each task (Pedregosa et al. 2011). The C value is optimized using grid search on the training dataset.

Sentence	Knowledge
General anesthesia increases abortion morbidity and mortality for women.	General anesthesia *positive*→ abortion morbidity
The U.N. treaty requires restrictions on tobacco advertising and sponsorship.	U.N. treaty *positive*→ restrictions on tobacco advertising
Overall, global warming will result in increased world rainfall.	global warming *positive*→ increased world rainfall
Stopping smoking decreases the risk of death by 18%.	Stopping smoking *negative*→ risk of death
For workers, privatization would mean smaller Social Security checks.	privatization *positive*→ smaller Social Security checks

Table 6: Examples of the automatically acquired knowledge.

Relation Detection, Relation Type Classification, and Consequence Classification Table 3 shows the results of using the framework for detecting the relation in terms of macro and micro F_1-score. The best performance is reached by applying the full set of features, which achieves 0.81 (micro) and 0.79 (macro) respectively. Lexical features perform by far best in distinguishing relations. According to the macro and micro F_1-scores, the results of classifying relation type seem similar to relation detection in the sense that the full set achieves the best scores (0.86 and 0.77 for micro and macro F1-score respectively). Again, the lexical feature is the most effective. Interestingly, the syntax features outperform the baseline only according to the macro score. Evaluating the pipeline of the relation detection along with the relation type classification in our framework, we achieve the following: a micro F_1-score of 0.77 and a macro F_1-score of 0.64. Regarding the classification of the consequence of a concept, as expected, the results of this task are low. However, the classifier still manages to outperform the baseline achieving 0.67 micro and 0.49 macro F_1-scores.

Table 4 shows the results of an ablation study regarding the features. Clearly, the lexical features shown to be, by a great deal, the most importnat for tackling the three tasks.

Concept Identification Based on the evaluation of the outputs of our OpenIE and SRL features, we found that the best feature is the OpenIE pattern of the main verb with [arg0] as the first concept and [arg1] as the second concept. The feature achieves an accuracy of 0.69 for identifying the first concept and 0.28 for the second concept.

Knowledge Acquisition for New Topics The framework achieves a reasonable effectiveness for knowledge acquisition when we apply it on our developed corpus. Yet, it is necessary to inspect its effectiveness at web scale.

To analyze whether the framework is effective enough to deal with new topics, we selected five topics that are not considered in our argumentation corpus from two high-quality resources: Wikipedia (the dump from April 23, 2019) and Annotated English Gigaword (a big corpus of newspaper articles). We downloaded both, extracted their articles and segmented the articles into sentences. Then, we identified all sentences that include one of the selected topics and applied our framework on them.

To examine the new extracted knowledge, we hire a professional editor (with Ph.D. degree, native English speaker) and asked him to check samples of the extracted knowledge. The samples were derived randomly after excluding knowledge that does not exceed certain thresholds, such as the classifier's confidence scores for the relation type and the confidence score of the concept grounding. The editor was provided with the two identified concepts and the relation between them, as well as with the sentence that the concepts and relation were extracted from. The editor had to read each sentence and check if the knowledge that we acquired is sound.

Table 6 shows statistics of the obtained knowledge with the evaluation results. Besides, we inspected the extracted knowledge manually. We found some cases that reveal the stance of a specific person on a certain topic. Also, we found knowledge that can be seen as historical information, especially extracted from Wikipedia, and knowledge which involves statistics from scientific studies. We found the knowledge identified in several sentences to be of a high quality.

Table 5 shows examples of the obtained knowledge. Such knowledge can be integrated into general knowledge-bases like ConceptNet (Speer, Chin, and Havasi 2017).

Conclusion

This paper studies what knowledge is needed for constructing argumentation knowledge graphs, how to represent that knowledge, and how to effectively acquire it. In particular, we have proposed a model for the positive and negative effect between concepts encapsulated in arguments and a methodology for identifying the modeled knowledge. To this end, we conducted a massive crowdsourcing annotation effort to annotate 4740 claims. We have then used the annotations to generate a new argumentation corpus and an argumentation knowledge graph. Also, we have developed a new framework with supervised learning and pattern-based approaches that automatically identifies the model's elements.

The proposed graph may serve as the heart of several computational argumentation applications, such as an exploratory search of arguments, an evaluation of the quality of arguments, and perhaps the detection of fake news. Also, it can be used for argumentative question answering systems, for example, to answer questions such as "What are the risks of nuclear energy?". In the future, we plan to build a large-scale web-based argumentation graph and investigate how we can exploit it for different computational argumentation tasks such as argument synthesis and argument search.

Acknowledgment

The paper is based on a project led by the first author during his internship at IBM Research. This work has been partially supported by the IBM PhD Fellowship Award.

References

Baccianella, S.; Esuli, A.; and Sebastiani, F. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC'10*.

Banerjee, S., and Pedersen, T. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *CICLing'02*, 136–145.

Bex, F.; Lawrence, J.; Snaith, M.; and Reed, C. 2013. Implementing the argument web. *Communications of the ACM* 56(10):66–73.

Botschen, T.; Sorokin, D.; and Gurevych, I. 2018. Frame- and entity-based knowledge for common-sense argumentative reasoning. In *5th Workshop on Argument Mining*, 90–96.

Choi, Y., and Wiebe, J. 2014. +/-effectwordnet: Sense-level lexicon acquisition for opinion inference. In *EMNLP'14*, 1181–1191.

Dasgupta, T.; Saha, R.; Dey, L.; and Naskar, A. 2018. Automatic extraction of causal relations from text using linguistically informed deep neural networks. In *SIGdial'18*, 306–316.

Dozat, T.; Qi, P.; and D. Manning, C. 2017. Stanford's graph-based neural dependency parser at the conll 2017 shared task. In *CoNLL'17 Shared Task*, 20–30.

Dunietz, J.; Levin, L.; and Carbonell, J. 2017. Automatically tagging constructions of causation and their slot-fillers. *TACL* 5:117–133.

Feng, V. W., and Hirst, G. 2011. Classifying arguments by scheme. In *ACL'11*, 987–996.

Ferragina, P., and Scaiella, U. 2010. Tagme: On-the-fly annotation of short text fragments (by wikipedia entities). In *CIKM'10*, 1625–1628.

Gemechu, D., and Reed, C. 2019. Decompositional argument mining: A general purpose approach for argument graph construction. In *ACL'19*, 516–526.

Guo, R.; Cheng, L.; Li, J.; Hahn, P. R.; and Liu, H. 2018. A survey of learning causality with data: Problems and methods. *arXiv preprint arXiv:1809.09337*.

Habernal, I.; Wachsmuth, H.; Gurevych, I.; and Stein, B. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *NAACL'18*, 1930–1940.

Hashimoto, C.; Torisawa, K.; Kloetzer, J.; Sano, M.; Varga, I.; Oh, J.-H.; and Kidawara, Y. 2014. Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In *ACL'14*, 987–997.

He, L.; Lee, K.; Lewis, M.; and Zettlemoyer, L. S. 2017. Deep semantic role labeling: What works and what's next. In *ACL'17*.

Hou, Y., and Jochim, C. 2017. Argument relation classification using a joint inference model. In *4th Workshop on Argument Mining*, 60–66.

Hua, X., and Wang, L. 2018. Neural argument generation augmented with externally retrieved evidence. In *ACL'18*, 219–230.

Huang, X.; Zhang, J.; Li, D.; and Li, P. 2019. Knowledge graph embedding based question answering. In *WSDM'9*, 105–113.

Kang, J. S.; Feng, S.; koglu, L.; and Choi, Y. 2014. Connotationwordnet: Learning connotation over the word+sense network. In *ACL'14*, 1544–1554.

Lawrence, J.; Visser, J.; and Reed, C. 2019. An online annotation assistant for argument schemes. In *Proceedings of the 13th Linguistic Annotation Workshop, LAW@ACL 2019*, 100–107.

Lebret, R.; Grangier, D.; and Auli, M. 2016. Neural text generation from structured data with application to the biography domain. In *EMNLP'16*, 1203–1213.

Mirza, P., and Tonelli, S. 2014. An analysis of causality between events and its relation to temporal information. In *COLING'14*, 2097–2106.

Moro, A.; Raganato, A.; and Navigli, R. 2014. Entity linking meets word sense disambiguation: a unified approach. *TACL* 2:231–244.

Napoles, C.; Gormley, M. R.; and Durme, B. V. 2012. Annotated gigaword. In *AKBC-WEKEX@NAACL-HLT'12*, 95–100.

Nguyen, D. Q.; Nguyen, T. D.; Nguyen, D. Q.; and Phung, D. 2018. A novel embedding model for knowledge base completion based on convolutional neural network. In *NAACL'18*, 327–333.

Pan, J.; Pavlova, S.; Li, C.; Li, N.; Li, Y.; and Liu, J. 2018. Content based fake news detection using knowledge graphs. In *ISWC'18*, 669–683.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* 12:2825–2830.

Raganato, A.; Camacho-Collados, J.; and Navigli, R. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *EACL'17*, 99–110.

Rashkin, H.; Sap, M.; Allaway, E.; Smith, N. A.; and Choi, Y. 2018. Event2mind: Commonsense inference on events, intents, and reactions. In *ACL'18*, 463–473.

Rashkin, H.; Singh, S.; and Choi, Y. 2016. Connotation frames: A data-driven investigation. In *ACL'16*.

Reisert, P.; Inoue, N.; Kuribayashi, T.; and Inui, K. 2018. Feasible annotation scheme for capturing policy argument reasoning using argument templates. In *5th Workshop on Argument Mining*, 79–89.

Saint-Dizier, P. 2016. Argument mining: the bottleneck of knowledge and language resources. In *LREC'16*.

Sap, M.; Le Bras, R.; Allaway, E.; dra Bhagavatula, C.; Lourie, N.; Rashkin, H.; Roof, B.; A Smith, N.; and Choi, Y. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *AAAI'19*.

Speer, R.; Chin, J.; and Havasi, C. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI'17*, 4444–4451.

Stab, C., and Gurevych, I. 2014. Identifying argumentative discourse structures in persuasive essays. In *EMNLP'14*, 46–56.

Stanovsky, G.; Michael, J.; Zettlemoyer, L. S.; and Dagan, I. 2018. Supervised open information extraction. In *NAACL'18*.

Stede, M., and Schneider, J. 2018. *Argumentation Mining*, volume 40 of *Synthesis Lectures in Human Language Technology*.

Toledo-Ronen, O.; Bar-Haim, R.; and Slonim, N. 2016. Expert stance graphs for computational argumentation. In *3th Workshop on Argument Mining*, 119–123.

Wachsmuth, H.; Naderi, N.; Hou, Y.; Bilu, Y.; Prabhakaran, V.; Thijm, T. A.; Hirst, G.; and Stein, B. 2017. Computational argumentation quality assessment in natural language. In *EACL'17*, 176–187.

Walton, D.; Reed, C.; and Macagno, F. 2008. *Argumentation Schemes*.

Walton, D. 2010. Types of dialogue and burdens of proof. In *Frontiers in Artificial Intelligence and Applications*, volume 216, 13–24.

Zhao, S.; Wang, Q.; Massung, S.; Qin, B.; Liu, T.; Wang, B.; and Zhai, C. 2017. Constructing and embedding abstract event causality networks from text snippets. In *WSDM'17*, 335–344.

Zhong, Z., and Ng, H. T. 2010. H.t.: It makes sense: A wide-coverage word sense disambiguation system for free text. In *ACL'10*, 78–83.