Unsupervised Stance Detection for Arguments from Consequences

Jonathan Kobbe, Ioana Hulpuș, Heiner Stuckenschmidt
University of Mannheim
{jonathan,ioana,heiner}@informatik.uni-mannheim.de

Abstract

Social media platforms have become an essential venue for online deliberation where users discuss arguments, debate, and form opinions. In this paper, we propose an unsupervised method to detect the stance of argumentative claims with respect to a topic. Most related work focuses on topic-specific supervised models that need to be trained for every emergent debate topic. To address this limitation, we propose a topic independent approach that focuses on a frequently encountered class of arguments, specifically, on arguments from consequences. We do this by extracting the effects that claims refer to, and proposing a means for inferring if the effect is a *good* or *bad* consequence. Our experiments provide promising results that are comparable to, and in particular regards even outperform BERT. Furthermore, we publish a novel dataset of arguments relating to consequences, annotated with Amazon Mechanical Turk.

1 Introduction

In the context of decision making it is crucial to compare positive and negative effects that result from a potential decision. Indeed, arguing for or against something because of its possible consequences is a frequent form of argumentation (Reisert et al., 2018; Al-Khatib et al., 2020). In this paper, we address the classical stance detection problem paying special attention to such arguments.

Stance detection, also called stance classification, is the task to decide whether a text is in favor of, against, or unrelated to a given topic. This problem is related to opinion mining, but while opinion mining focuses on the sentiment polarity explicitly expressed by a text, stance detection aims to determine the position that the text holds with respect to a topic that is generally more abstract and might not be mentioned in the text. As such, in stance detection, texts can transmit a negative sentiment or opinion, but be in favor of the targeted topic. For example, the text *Holocaust denial psychologically harms Holocaust survivors* expresses a negative opinion, but its stance towards *Criminalization of Holocaust denial* is positive.[1]

Recently, the problem of stance detection has received growing attention from the scientific community, as shown by the recent survey of Küçük and Can (2020). Most approaches tackle this problem by learning stance classification models for each topic. While this can achieve good results, new models need to be trained for each new topic of interest, generally entailing large annotation studies.

While we admit that a one-size-fits-all approach to stance detection is currently unfeasible, we take a different perspective. Rather than targeting topic-dependent models, we target a subclass of arguments. Specifically, we focus on arguments that have been classified by Walton et al. (2008) under the *argument from consequences* scheme. They contain a premise of the form *If A is brought about, then good (bad) consequences will (may plausibly) occur*, and a conclusion *A should (not) be brought about*. In most real-life arguments of this type, the consequences are expressed, but the interpretation that they are *good* or *bad*, as well as the conclusion, are most often implicit. The task of stance detection is then to determine if the argument is against or in favor of *A*. Our solution to find the stance of such arguments revolves around extracting and analyzing cause-effect relations in order to infer if the consequences are *good* or *bad*.

We conducted an Amazon Mechanical Turk (AMT) study, in which we crowdsourced annotations for 1894 arguments extracted from Debatepedia. We compared our system's performance

[1] All arguments presented in this paper are from http://www.debatepedia.org.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 50–60,
November 16–20, 2020. ©2020 Association for Computational Linguistics

to a sentiment analysis baseline and a fine-tuned BERT model. The results show that our results are comparable and, in some settings, even better than BERT's.[2] Aside from not needing annotated training data, we stress the advantage of our approach for providing human-understandable explanations to the results, and to provide, as a by-product, cause-effect relations between concepts brought up in arguments.

The paper is structured as follows. Section 2 positions our contributions with respect to related literature. Section 3 presents our proposed approach. Section 4 describes our crowdsourced dataset, which we use in Section 5 to evaluate our approach. Lastly, Section 6 concludes the paper.

2 Related Work

Stance detection has been studied on various types of formal texts such as congressional debates (Thomas et al., 2006) and company-internal discussions (Murakami and Raymond, 2010). However, like most recent related work on the topic, we are particularly interested in informal texts from online social media.

The vast majority of previous approaches proposes supervised methods, using traditional machine learning algorithms (Somasundaran and Wiebe, 2010; Anand et al., 2011; Hasan and Ng, 2013; Faulkner, 2014; Sobhani et al., 2016; Addawood et al., 2017) and more recently, various deep neural networks architectures (Sun et al., 2018; Du et al., 2017; Dey et al., 2018; Ghosh et al., 2019). These approaches, most of which have been triggered by a recent SemEval shared task[3] (Mohammad et al., 2016), learn topic-specific models. Thus, new topics require new models whose training entails large user annotation studies. In contrast, we propose a fully unsupervised, topic-independent method, and rather target a particular but frequent class of claims, those that refer to consequences.

Among the unsupervised approaches, the most prominent one is this of Somasundaran and Wiebe (2009), which got extended by Konjengbam et al. (2018) and Ghosh et al. (2018). However, they focus on non-ideological topics (usually products, e.g., *iPhone vs. Galaxy*). In contrast, we target ideological topics (e.g., *Gay Marriage, Abortion*) whose stance is harder to detect due to less frequent use of sentiment words and a wider variety of brought up issues and arguments (Rajendran et al., 2016; Wang et al., 2019). On the one hand, these works extract topic aspects (e.g., *screen resolution, battery*) and polarities towards these aspects, a step that is unfeasible for ideological topics. On the other hand, like these works, we also use syntactic rules, but not for pairing aspects to opinions, but for extracting triples that correspond to statements about effects over opinion words.

Another class of stance detection approaches uses the context of the post, such as its relations to other posts in the debate, the network of authors, or the author's identity (Hasan and Ng, 2013; Sridhar et al., 2014; Addawood et al., 2017; Bar-Haim et al., 2017b). By contrast, we target claim-topic pairs in isolation.

Another aspect that sets our work apart from most related work is that, except for the approaches that target tweets, most focus on longer texts while we consider short, one-sentence claims. In this regard, but not only, the stance detection work that is closest to ours is the partly supervised system of Bar-Haim et al. (2017a). They also propose a topic-independent solution to stance detection for short claims without considering context, but they do not specifically address arguments from consequences. While they follow a similar sequence of steps as we do, they propose different approaches for each step. For instance, they propose a supervised approach to detect the target of a claim's opinion, while we do it in an unsupervised manner. They focus primarily on detecting contrastive relations between phrases, while our focus is on detecting effects. In this last regard, the works can be considered complementary.

Regarding the analysis of arguments from consequences, Reisert et al. (2018) provide and use scheme dependent templates to analyze the structure of arguments. Their work is rather conceptual and focuses on annotations. Very recently, Al-Khatib et al. (2020) built, on similar intuitions as ours, an approach for creating argumentation knowledge graphs based on cause-effect relations. Their work comes to reinforce the usefulness of addressing arguments from consequences.

To sum up, our contribution is three-fold: (i) we propose a fully unsupervised approach for stance detection, focusing on arguments that refer to consequences; (ii) we define rules over grammatical dependencies that exploit sentiment as well as ef-

[2] Our data and source code are publicly available at https://github.com/dwslab/StArCon.

[3] http://alt.qcri.org/semeval2016/task6

fect words in order to determine *good* and *bad* consequences; (iii) we publish a new stance detection dataset that labels claims that refer to consequences, and which was crowdsourced on AMT.

3 Our Approach

Given an argumentative claim and a topic, our task is to detect the stance that the claim has with respect to the topic. Statements such as the claim or topic usually express a positive (favorable) or negative (unfavorable) position to a concept that we call the **target**. As such, the target is a phrase that belongs to the statement. In the example shown

Topic:	Medical marijuana dispensaries
Claim:	Legalizing medical marijuana does not increase use and abuse

Table 1: Example of topic-claim pair

in Table 1, the target of both topic and claim is *medical marijuana*. Our solution starts by first determining the stance of the claim and of the topic towards their respective targets T_c and T_t. We then use these stances and the semantic relation between the targets to determine the claim's stance towards the topic.

The overarching intuition behind our approach is that when the stance of a statement towards its target is favorable, the text either highlights the desirable consequences of the target being brought about (e.g., *Electing an EU president directly will increase accountability*), or it highlights the negative consequences if the target is not brought about (e.g., *Sinking organic blooms can render the deep sea anoxic*).

At the core of our approach resides what we call the **effect triple**. The effect triple is a triple of the form $< (T, dir), (P, \mathit{eff}), (O, sent) >$. The (T, dir) pair represents the target T of the statement and if the statement refers to a magnification ($dir = 1$) (e.g. *legalizing medical marijuana*), or a reduction ($dir = -1$) of the target (e.g. *banning medical marijuana*). The (P, eff) pair represents the predicate P that has T as the subject, together with the effect eff that it has over the object O. The effect can be positive ($\mathit{eff} = +1$) or negative ($\mathit{eff} = -1$). Lastly, the $(O, sent)$ pair represents the object over which T has the effect P. We expect the *sentiment* of an object to reflect whether it is generally regarded as *a good thing* ($sent = +1$) or *a bad thing* ($sent = -1$).

Our approach's core idea is to distill such an effect triple from the claim and use it to infer the claim's stance towards T_c. We further determine (T_t, dir) to infer the topic's stance towards T_t. Using these stances, together with the relation between the claim's and the topic's target, we finally decide the claim's stance with respect to the topic. We now describe the lexicons we use as well as each of these steps in more detail.

3.1 Lexicons

For determining dir, eff, and $sent$, we use an effect verb lexicon and a sentiment lexicon that we describe in the following.

The ECF Effect Lexicon To identify verbs and nominalized verbs that indicate effects on their direct objects, we extend the connotation frames (Rashkin et al., 2016). The connotation frames lexicon consists of a list of 947 verbs, manually annotated with values in the $[-1, 1]$ range, indicating if the verb implies a positive or negative effect over its object. We consider the entries with scores in the range $[-0.1, 0.1]$ as a neutral effect (e.g., *use, say, seem*), and we filter them out. We call the 845 remaining words in the lexicon **effect words**. We extend the list of effect words by adding all words in the same WordNet (Fellbaum, 2010) synset as the effect words, as long as there is no contradiction. A contradiction occurs when a new candidate effect word shares a synset with both a negative and a positive effect word. This way, we obtain 2508 effect words. We call this lexicon the extended connotation frames lexicon (ECF). As ECF only contains verbs, we use it via the stems of the words, mainly to also get the effects of nominalized verbs. In our experiments, we compare the performance of this lexicon with +/-EffectWordNet (Choi and Wiebe, 2014)(EWN).

The Sentiment Lexicon In order to determine if the object of the effect is something *good* or *bad*, we combine several commonly used sentiment lexicons: (i) the MPQA lexicon[4] (Wilson et al., 2005), (ii) the opinion lexicon of Hu and Liu (2004), and (iii) the sentiment lexicon of Toledo-Ronen et al. (2018) (uni- and bigrams, using a threshold of ± 0.2). The composed lexicon contains sentiment values in the range $[-1, 1]$.

[4]We used an American English dictionary to correct orthographic mistakes resp. to add American English versions of British English words.

For many words, the polarities of their sentiment and of their effect are the same (e.g., *kill, love*). Still, there are important exceptions, such as *reduce*, which has neutral sentiment but indicates a negative effect, or *conquer*, which has a slightly positive sentiment but indicates a negative effect.

3.2 Effect Triple Extraction

Target Identification To detect the targets of the claim (T_c) and topic (T_t), we assume that T_c is semantically related to the topic, or more specifically, to T_t. Thus, we identify T_c and T_t simultaneously by following three strategies. The use of the second and third strategies is conditioned on the previous strategies to have failed to identify a pair of targets. First, we look for a pair of nouns that are identical or have the same lemma. We use Stanford Core NLP (Manning et al., 2014) for POS tagging and lemmatizing. Second, we look for a pair consisting of an acronym (e.g., *ICC*) and a word sequence whose first letters form the acronym (e.g., *International Criminal Court*). Third, we look for pairs of nouns that are synonyms or antonyms according to *Thesaurus.plus*[5].

Besides returning T_c and T_t, we also return a value $r = +1$ if the two targets have been found to be synonyms and $r = -1$ if they are antonyms. Thus, first and second strategies only return $r = 1$ while the third strategy returns 1 or -1.

Target Direction Determination As described earlier, each target is accompanied by a dir value which indicates if the statement refers to a phenomenon of amplification or reduction of the target. We detect this by searching for a word whose object is the target by using Patterns 1 and 2 shown in Table 2. The word is then looked-up in the effect lexicon. If a negative effect is found, then $dir = -1$, otherwise $dir = 1$. We call the word the *target effector*, or just *effector*. In the claim in Table 1, the effector is *legalizing* and expresses an amplification of the target ($dir = 1$).

Detecting Predicates and Their Effects Effect words are commonly used in arguments from consequences to express a (potential) effect that the target has or might have over another object. For example, in the claim in Table 1, the effect word *increase* expresses a positive effect that the (amplified) target has over the objects *use, abuse*.

We detect this effect of the target by using Pattern 3 to find a predicate whose subject is either the target or its effector, and by looking up this predicate in the effect lexicon. We thereby set eff to 1 or -1, depending on if the effect is positive or negative. In our running example, the (P, eff) pair becomes $(\mathit{increase}, -1)$ because of the negation, as we explain below.

Telling *good* from *bad* The last effect triple component we detect is $(O, sent)$. To this end, we search the dependency graph for instantiations of Patterns 1 or 2, where P is the predicate that has been detected to express the target's effect. If such an object is found, we use the sentiment lexicon by first searching for the exact word and, if not available, for the word's lemma. We set $sent$ to -1 if the word bears a negative sentiment or to 1 otherwise. In our example, the $(O, sent)$ pair becomes $(abuse, -1)$ because the word *use* is neutral per se.

The sentiment of a word is overwritten by the sentiment of its modifiers, as shown in Pattern 4 in Table 2. In the provided example in the table, one can see that the modifier *terrorist* dominates the sentiment of the positive word *haven*. Consequently, both *terrorist haven* and *terrorist attack* are considered generally bad.

Negation We deal with negations for each effect triple component. We identify negations by looking for Patterns 5, 6, and 7, as shown in Table 2. Patterns 5 and 6 make use of a manually created list of all negative English prepositions[6]. The existence of a negation affecting the target, predicate, or object toggles the sign of the corresponding value - dir, eff or $sent$, respectively.

3.3 Inferring the Stance Towards the Target

To infer the stance that a statement expresses towards its target, we use the intuition that the stance is unfavorable when the text expresses negative consequences of the target, and positive otherwise. Thus, we define that the stance towards the target is positive in exactly the following four cases: (i) the target's amplification implies a positive effect over something good ($dir = \mathit{eff} = sent = +1$); (ii) the target's amplification implies a negative effect over something bad ($dir = +1$, $\mathit{eff} = sent = -1$); (iii) the target's reduction implies a negative effect over something

[5]We use only the synonyms and antonyms shown at `https://thesaurus.plus/thesaurus/xxx` where `xxx` is a placeholder for concrete words

[6]Those are *except, less, minus, opposite, sans, unlike, versus, without, w/o, vice, instead (of), lack.*

	Pattern	*Interpretation*	*Example*
1	$P \xrightarrow{*} O$	P has object O	Insurance mandates violate the rights of employers. dobj
2	$P \xrightarrow{prep}? \xrightarrow{pobj} O$	P has object O	The military industrial complex profits from escalation in Afg. prep pobj
3	$P \xrightarrow{\diamond} S$	P has subject S	Holocaust denial is inherently descriminatory and damaging. nsubj
4	$X \xrightarrow{\dagger} M, sent(M) \neq 0$	$sent(X) := sent(M)$	W/o more troops, Afgh will become terrorist haven amod
5	$NegP \xrightarrow{pobj} X$	X is negated	Free speech without Fairness Doctrine can harm policy-making pobj
6	$X \rightarrow NegP, \nexists NegP \xrightarrow{pobj}$	X is negated	W/o more troops, Afgh will become terrorist haven nn
7	$X \xrightarrow{neg}$	X is negated	Solar energy does not damage air quality. neg

Table 2: Dependency graph patterns. $* \in \{dobj, nsubjpass, cobj, csubjpass, nmod, xcomp\}$; $\diamond \in \{nsubj, csubj\}$; $\dagger \in \{amod, nn, advmod\}$; $NegP$ stands for *negative preposition*

good (dir = *eff* $= -1$, $sent = +1$); (iv) the target's reduction implies a positive effect over something bad ($dir = +1$, *eff* $= -1$, $sent = +1$). Hence, the stance is favorable towards the target if the multiplication of the three components' values is $+1$. Consequently, we define the stance of a statement towards the target as $s = dir \cdot eff \cdot val$ and interpret $s = 1$ as *In favor* and $s = -1$ as *Against*.

3.4 Inferring the Stance of the Claim Towards the Topic

The steps above can be executed analogously for the claim and the topic. However, due to the nature of the text expressing the topic, we only aim to extract an effect triple from the claim. For the topic, we detect its target and set the stance to its corresponding dir value. We denote the stances of the claim and topic towards their respective targets as s_c and s_t. To infer the claim's stance towards the topic, we need to consider the relation between T_c and T_t, i.e., the value of r as described in Section 3.2. We then define the final result of the analysis as $\Pi = s_c \cdot s_t \cdot r$.

Table 3 presents further examples of how our approach detects the stance of the claim towards the topic. As illustrated in the examples, the straightforward interpretability of the stance detection process can be easily used for producing human-readable explanations for the returned results. This is particularly relevant for helping users get more control over the process, particularly in light of subsequent applications on top of stance detection.

	Porn watching may actually reduce rape rates	**Pornography**
T, dir P, *eff* O, $sent$	*Porn*, $+1$ *reduce*, -1 *rape rates*, -1	*Pornography*, $+1$
s	1	1
r	1	
Π	1 (In favor)	

	Holocaust denial psychologically harms Holocaust survivors	**Criminalization of Holocaust denial**
T, dir P, *eff* O, $sent$	*Holocaust denial*, 1 *harms*, -1 *survivors*, $+1$	*Hol. denial*, -1
s	-1	-1
r	1	
Π	1 (In favor)	

Table 3: Worked out Examples

3.5 Alternative Strategies

We denote the process in which all the previous steps are fulfilled and an effect triple is extracted as **TPO**. However, due to a variety of reasons that we analyze in Section 5.4, we might fail to extract a complete effect triple. One such case is when an adjective expresses an effect, for instance, *Holocaust denial is discriminatory*. For that reason, if we identify T and P, but not O, we set *eff* to the sentiment polarity of P, and $sent$ to $+1$ by default. We refer to this strategy as **TP**.

Another potential situation is that the system detects (P, eff) and $(O, sent)$, but it can not relate them to T. One cause can be that we fail to identify T. If so, $dir = +1$ by default. Another cause can be that T is found, but we can not infer its relation to P. In this case, we consider that the

identified target is the subject of P and set (T, dir) accordingly. We refer to this strategy as **PO**.

Lastly, if all above strategies fail to create an effect triple, we use a heuristic: if T was found, dir is set accordingly. Otherwise $dir = 1$ by default. For the remaining words in the statement, we check their sentiment score, still using Pattern 4, toggling the sign if it is negated. The sum of the sentiment scores is then multiplied with dir. The stance is considered favorable or not depending on the sign of the result. We refer to this strategy as **Heuristic**.

4 Dataset Generation

To evaluate our approach, we need stance annotated topic-claim pairs, as well as annotations if the topic-claim pair refers to a consequence or not.

4.1 Data Collection

To create such a corpus, we run an AMT crowdsourcing study, where we annotate claims and topics extracted from Debatepedia[7]. We only use the 236 *Featured Debate Digest articles* as they are of higher quality. They contain more than 10,000 arguments labeled by their author as either pro or con the debate's topic. Usually, the arguments start with a bolded, one-sentence summary, which serves as the argument's claim. We exclusively use these claims and pair them to the debate's topic. We exclude 16 debates whose topics contain *vs* or *or* (e.g. *Democrats vs. Republicans*), and 30 debates without a title question. To create a balanced dataset that covers a large variety of topics, we randomly selected 5 pro and 5 con arguments of each debate. If a debate contains less than 5 pro and 5 con arguments, we select the maximum equal number of pro and con arguments. We obtain 190 different topics and 1894 arguments.

4.2 Crowdsourcing Study

The annotation task consisted of the debate's topic, one of its claims, and two questions. The first question was to select the stance of the claim towards the topic, out of the following choices: *in favor, against, neither* and *I don't know*. Although we have the original arguments' stances, this question helps us check how clear the claim is when taken out of the debate's context. The second question was whether the claim refers to a consequence related to the topic, with possible answers *yes, no* and *I don't know*. Each topic-claim pair was annotated by 10 annotators living in the US with a HIT approval rate greater than 98% and more than 10,000 approved HITs in total. Overall, 277 annotators worked on the task.

[7] http://www.debatepedia.org

Valid Annotations	*Stance* rate	κ	κ'	*Consequence* rate	κ	κ'
6	.002	-.10	-.20	.001	-.17	-1
7	.013	.11	.15	.008	.04	.10
8	.051	.24	.32	.036	.06	.24
9	.183	.34	.58	.207	.23	.44
10	.751	.52	.74	.748	.25	.58
Weight. Avg		.47	.68		.24	.53

Table 4: Fleiss' Kappa dependent on the number of valid annotations

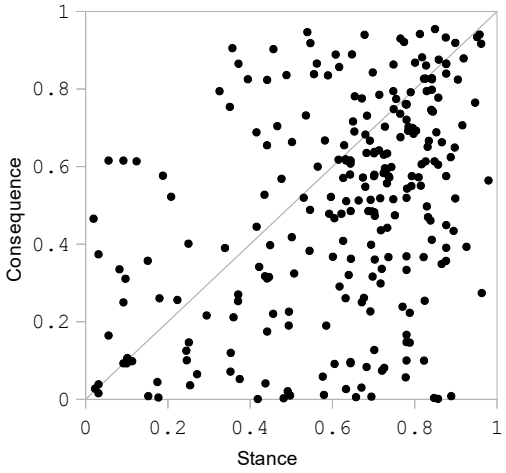

Figure 1: Reliability of annotators according to MACE: The higher the score, the more reliable the annotator is.

4.3 Agreement and Reliability

Table 4 shows the inter-annotator agreement per number of valid annotations, i.e., annotations that are not *I don't know*. Since we have many annotators, Fleiss κ is particularly low on consequence annotation, but still indicates higher agreement than random. To give an agreement estimate less sensitive to individual outliers, we also compute κ' as the Fleiss kappa between two "experts", where each expert brings together half of the number of annotators and its annotation is decided with MACE (Hovy et al., 2013).

Figure 1 shows the reliability of individual annotators. Although there is a weak correlation among the reliability of the two tasks (Pearson .41), some annotators are quite reliable in annotating stances, but highly unreliable in annotating consequences. This indicates that the latter task was unclear to some of the annotators. To understand why the annotators usually disagree, we investigated such

instances and identified several possible reasons:

Complexity In the topic-claim pair *Criminalization of Holocaust denial – Danger of public accepting holocaust denial should be fought by logic*, both topic and claim have a negative stance towards *holocaust denial*, which suggests the label *in favor*. Still, by proposing a different solution than *criminalization*, the claim is *against* the topic.

Missing Background Knowledge Many arguments involve non-trivial background knowledge: *Israeli military assault in Gaza – Hamas was first to escalate conflict following end of ceasefire*.

Ambiguity According to the pair *2009 US economic stimulus – Stimulus risks being too small not too large*, a small stimulus is bad while an appropriate stimulus is good.

Ethical Judgement Different judgments on what is good and bad can lead to different stance labels: *Ban on human reproductive cloning – Cloning will involve the creation of children for predetermined roles*.

Lack of Conceptual Clarity Especially deciding whether the claim refers to a consequence related to the topic can be a matter of judgment. For example, in *Health insurance mandates – Insurance mandates violate the rights of employers*, the violation of rights can be seen as a consequence or as a property of insurance mandates.

4.4 Final Dataset

To account for unreliable annotators, we compute the annotation result with MACE. As such, we find that for 81.36% of the annotated arguments, the stance label obtained via MACE is the same as the original stance label. By comparison, the majority vote matches 79.30% of the original stance labels. Since disagreements between the MACE annotation and the original stance might indicate that the claim's stance is unclear outside the debate's context, we exclude from the dataset all such pairs. For example, the original label of the pair *Is Wikipedia valuable? – Wikipedia is online and interactive, unlike other encyclopedias* is *con*, because, in its context, it was discussed whether Wikipedia is an encyclopedia or not. In contrast, the result of our annotation is *pro*. Since the original labels are only *pro* or *con*, all pairs that our study determined as *neither* are removed. This filter resulted in a total of 1502 pairs, out of which 822 have been annotated to relate to consequences.

conseq		*other*		*debate*		*wiki*	
pro	con	pro	con	pro	con	pro	con
376	446	370	310	746	756	1195	1199

Table 5: Class distributions

5 Evaluation

5.1 Data

We report results both on the 822 pairs that relate to consequences, denoted by *conseq*, and on the rest of the pairs, denoted by *other*, as well as on their union, denoted by *debate*.

For checking the performance of the systems on an independent dataset, we also use the claim stance dataset[8] published by Bar-Haim et al. (2017a). This dataset contains 55 topics of *idebate*[9] and 2394 manually collected claims from Wikipedia. We denote this dataset by *wiki*. As Bar-Haim et al. (2017a,b) do, when working with this dataset, we use only the topic's target and not the entire topic to ensure comparability.

Table 5 shows the class distribution of the datasets.

5.2 Compared systems

We evaluate our system with the effect lexicon lexicon that we describe in Section 3.1 (**ECF**), as well as with the +/-EffectWordNet (**EWN**). For comparison, we implement two other approaches:

sent As a baseline, we use a system that simply sums up all the sentiment scores in the claim. For the *wiki* dataset, the sign is switched if the topic sentiment is negative.

BERT As state of the art, we use BERT (Devlin et al., 2019), which was recently shown to outperform a series of alternative stance detection systems (Ghosh et al., 2019). We fine-tune BERT using the large, uncased pre-trained weights.[10] Just as Schiller et al. (2020), we set the number of epochs to 5 and the batch size to 16. The input are topic-claim pairs. We perform 10-fold cross-validation with a train-dev-test ratio of (70/20/10), ensuring that each topic exclusively occurs in one set.

5.3 Results and Discussion

The results that compare our system to BERT and the sentiment detection baseline are presented in

[8] Available at https://www.research.ibm.com/haifa/dept/vst/debating_data.shtml

[9] https://idebate.org/

[10] We worked with the original release: https://github.com/google-research/bert

	conseq				other				debate				wiki			
	pro	con	mac	acc	pro	con	mac	acc	pro	con	mac	acc	pro	con	mac	acc
sent	.62	.67	.65	.65	.64	.47	.56	.57	.63	.59	.61	.61	.61	.58	.60	.60
BERT	.65	**.82**	**.74**	**.78**	**.73**	.48	.60	**.66**	.63	**.72**	.67	**.71**	**.72**	**.65**	**.68**	**.70**
- BERT std deviation	*.33*	*.08*	*.20*	*.13*	*.06*	*.31*	*.17*	*.11*	*.32*	*.18*	*.21*	*.15*	*.07*	*.24*	*.15*	*.11*
our system ECF	**.72**	.74	.73	.73	.69	**.56**	**.63**	.64	**.71**	.67	**.69**	.69	.66	.63	.64	.64
our system EWN	.70	.72	.71	.71	.66	.53	.60	.61	.68	.64	.66	.66	.64	.61	.63	.63

Table 6: Experimental results. F1 scores per stance class (*pro* and *con*), macro-F1 (*mac*), and Accuracy (*acc*). For *BERT*, we show the mean of the respective cross-validation results and their standard deviation.

Table 6. First, as expected, our system performs better on arguments related to consequences than on other arguments, with a macro-F1 difference of 10pp between *conseq* and *other*. Further, our system with both lexicon settings consistently outperforms the *sent* baseline, but its macro-F1 score is outperformed by BERT on *conseq* and *wiki*, and its accuracy is outperformed by BERT on all datasets. This is not surprising, given that we use BERT pre-trained and then fine-tuned to our data. Interestingly, our system with ECF achieves better results than BERT in terms of macro F1 score on the arguments that are *not* related to consequences (*other*), and on the complete *debate* dataset. This indicates that our method can deal reasonably well with arguments that are not from consequences.

Concerning the two stance classes, with both lexicon settings, our system is better than BERT at predicting the *pro* class in arguments from consequences, but is outperformed on the *con* class. Another interesting result is that on *conseq*, our system has a quite similar performance on the *pro* and *con* classes with both lexicon settings . In contrast, BERT's performance varies drastically, with a difference of approximately 17pp in favor of the *con* class. BERT's high variability is also indicated by the high standard deviation on the 10 folds. For comparison, we also computed the F1 macro standard deviation of our system with ECF when run on the same 10 folds, and the values lie between .03 on *debate* and .07 on *conseq*. This indicates that our unsupervised approach is more robust with more predictable performance.

Concerning the two effect lexicons, our system performs consistently better when using ECF than when using EWN. Our analysis indicates that the high coverage of the EWN lexicon comes at the expense of accuracy. Therefore, in the following, we will only refer to our system using ECF.

Regarding the two datasets *debate* and *wiki*, BERT outperforms our system, with quite a high margin particularly on the *wiki* data. The accuracy that Bar-Haim et al. (2017a,b) report on the *wiki* data, when no context features are used, is .68 which is lower than BERT's (.70) but higher than ours (.65 for evaluating on the dedicated test set). This is not surprising given that the data contains general arguments. Nevertheless, as our approach only targets a subclass of these arguments, the results are quite promising. Unfortunately, Bar-Haim et al. (2017a,b)'s system is proprietary and we could not evaluate it on our *conseq* data.

	conseq		other		debate		wiki	
	r	F1	r	F1	r	F1	r	F1
Total	1	.73	1	.63	1	.69	1	.64
Target found	.82	.74	.76	.64	.80	.70	.53	.67
-Word/Lemma	.75	.74	.72	.64	.74	.70	.42	.67
-Acronym	.02	.80	.01	.89	.02	.83	.00	–
-Syn/Ant	.05	.69	.03	.50	.04	.64	.11	.66
TPO/TP/PO	.60	.76	.39	.64	.51	.72	.54	.67
-TPO	.23	.74	.05	.65	.15	.73	.07	.81
-TP	.21	.84	.18	.74	.20	.80	.10	.77
-PO	.16	.69	.16	.53	.16	.62	.36	.62
Heuristic	.40	.68	.61	.61	.49	.65	.46	.61

Table 7: Evaluation of the target identification and stance detection strategies; r denotes the rate of data instances.

Table 7 provides further insights into our solution. First, on all Debatepedia based datasets, we find a target in more than .75 of the data instances, and overall, the results are slightly better when a target is found. Most of the targets are found by word similarity and the fewest by the acronym. The results obtained on the instances where the target was found by synonym/antonym relations are significantly lower than those obtained when the target was found with the other two strategies. This indicates that the approach is sensitive to semantic drift in target identification.

Overall, we identify a potential consequence (*TPO/TP/PO*) for .6 of the arguments in *conseq*.

While the results are quite good on all datasets when we detect a complete effect triple (*TPO*), they are overtaken by results of the *TP* cases. Together, the instances solved with *TPO* and *TP* strategies amount to .44 of the *conseq* dataset but to much lower on the other datasets (e.g., only .17 on the *wiki*). The performance on the *PO* cases is comparable to the performance on the *Heuristic* cases, and significantly lower than when *TPO* or *TP* could be applied. Depending on the dataset, the system needed to apply the Heuristic strategy on .4 to .61 of the instances. Our efforts for future work are directed towards helping the system make sense of more of the claims so that the number of times it needs to fallback to *PO* and *Heuristic* are reduced.

5.4 Error Analysis

To better understand the limitations of our approach, we analyzed the errors on the *conseq* data and found several reasons for wrong predictions:

Incomplete list of patterns Some arguments cannot be meaningfully analyzed with our current list of patterns. We plan to extend this list with more complex patterns, while we are also working on automatically learning such patterns from data.

Conceptual errors We assume that positive effects on something negative result in something negative (e.g., *War in Iraq has helped terrorist recruitment.*). However, this is not always the case (e.g., *Privatizing social security helps the poor.*).

Finding the targets As shown in Table 7, we often fail to detect targets. For example, our target detection strategies fail on the claim-topic pair *Standardized tests ensure students learn essential information. – No Child Left Behind Act*. In this specific case, there is a hypernym relation between the topic and *Standardized tests*. Further, we found that our straightforward approach to identifying targets and the relations between them is one of the core reasons for our approach's poorer performance on the *wiki* data compared to the *debate* data. Improving the target finding strategy by leveraging additional semantic knowledge is one of the core directions for our future work.

Missing / wrong lexicon entries For many words, we are missing an entry in our lexicons, or the entry exists but is questionable. For instance, in the sentiment lexicon, *Palestinian* is annotated with a negative sentiment. Also, sometimes the effect on the object seems to be mixed up with the word's overall effect. For example, *solve* has a positive effect on the object in both ECF and EWN lexicons, but arguably when a problem is *solved*, it undergoes a reduction (e.g. *Reforestation,[...] can help solve global warming*).

Ambiguity Some words have a positive or negative effect depending on the sense with which they are used (e.g., *push* vs. *push for*). In the effect lexicon, we have only one entry per word. In the EWN, there are multiple senses, but we always use the most probable effect. Word sense disambiguation is required for these cases, which is known to be very challenging for verbs. However, a potential solution could be to annotate VerbNet frames with effects, but this is outside the scope of this work.

Text parsing errors As our method relies on the output of the dependency parser, the Lemmatizer, the POS tagger, and the Stemmer, their errors naturally propagate.

6 Conclusion and Future Work

We propose a fully unsupervised method to detect the stance of arguments from consequences in online debates. The method exploits grammatical dependencies and lexicons to identify effect words and their impact. For our evaluation, we annotated arguments from *Debatepedia* regarding their stance and whether they involve consequences or not. The results we obtained are motivating. Our method is comparable to BERT while being more robust.

Besides the future extensions of this approach that we mentioned in our results discussion and error analysis, this work opens several interesting research paths. Mainly, its good performance on the claims that refer to consequences reinforces our intuition that designing systems tailored for particular argumentation schemes might be a good alternative to topic-specific models. Therefore, we plan to complement this work with approaches for other frequently applied schemes such as *arguments by expert opinion* and *arguments by example*.

Acknowledgments

This work has been funded by the Deutsche Forschungsgemeinschaft (DFG) within the project ExpLAIN, Grant Number STU 266/14-1, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

References

Aseel Addawood, Jodi Schneider, and Masooda Bashir. 2017. Stance classification of twitter debates: The encryption debate as a use case. In *Proceedings of the 8th International Conference on Social Media & Society*, #SMSociety17, New York, NY, USA. Association for Computing Machinery.

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 Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2020)*.

Pranav Anand, Marilyn Walker, Rob Abbott, Jean E. Fox Tree, Robeson Bowmani, and Michael Minor. 2011. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2.011)*, pages 1–9, Portland, Oregon. Association for Computational Linguistics.

Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. 2017a. Stance classification of context-dependent claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 251–261, Valencia, Spain. Association for Computational Linguistics.

Roy Bar-Haim, Lilach Edelstein, Charles Jochim, and Noam Slonim. 2017b. Improving claim stance classification with lexical knowledge expansion and context utilization. In *Proceedings of the 4th Workshop on Argument Mining*. Association for Computational Linguistics.

Yoonjung Choi and Janyce Wiebe. 2014. +/-EffectWordNet: Sense-level lexicon acquisition for opinion inference. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1181–1191, Doha, Qatar. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. 2018. Topical stance detection for twitter: A two-phase lstm model using attention. In *Advances in Information Retrieval*, pages 529–536, Cham. Springer International Publishing.

Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. 2017. Stance classification with target-specific neural attention. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3988–3994.

Adam Faulkner. 2014. Automated classification of stance in student essays: An approach using stance target information and the wikipedia link-based measure. *Proceedings of the 27th International Florida Artificial Intelligence Research Society Conference, FLAIRS 2014*, pages 174–179.

Christiane Fellbaum. 2010. Princeton university: About wordnet.

Shalmoli Ghosh, Prajwal Singhania, Siddharth Singh, Koustav Rudra, and Saptarshi Ghosh. 2019. Stance detection in web and social media: A comparative study. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 75–87, Cham. Springer International Publishing.

Subrata Ghosh, Konjengbam Anand, Sailaja Rajanala, A Bharath Reddy, and Manish Singh. 2018. Unsupervised Stance Classification in Online Debates. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, CoDS-COMAD '18, pages 30–36, New York, NY, USA. ACM. Event-place: Goa, India.

Kazi Saidul Hasan and Vincent Ng. 2013. Extra-linguistic constraints on stance recognition in ideological debates. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 816–821, Sofia, Bulgaria. Association for Computational Linguistics.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA. Association for Computing Machinery.

Anand Konjengbam, Subrata Ghosh, Nagendra Kumar, and Manish Singh. 2018. Debate stance classification using word embeddings. In *Big Data Analytics and Knowledge Discovery*, pages 382–395, Cham. Springer International Publishing.

Dilek Küçük and Fazli Can. 2020. Stance detection: A survey. *ACM Comput. Surv.*, 53(1).

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California. Association for Computational Linguistics.

Akiko Murakami and Rudy Raymond. 2010. Support or oppose? classifying positions in online debates from reply activities and opinion expressions. In *Coling 2010: Posters*, pages 869–875, Beijing, China. Coling 2010 Organizing Committee.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2016. Contextual stance classification of opinions: A step towards enthymeme reconstruction in online reviews. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 31–39, Berlin, Germany. Association for Computational Linguistics.

Hannah Rashkin, Sameer Singh, and Yejin Choi. 2016. Connotation Frames: A Data-Driven Investigation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 311–321. Association for Computational Linguistics. Event-place: Berlin, Germany.

Paul Reisert, Naoya Inoue, Tatsuki Kuribayashi, and Kentaro Inui. 2018. Feasible Annotation Scheme for Capturing Policy Argument Reasoning using Argument Templates. In *Proceedings of the 5th Workshop on Argument Mining*, pages 79–89, Brussels, Belgium. Association for Computational Linguistics.

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2020. Stance detection benchmark: How robust is your stance detection?

Parinaz Sobhani, Saif Mohammad, and Svetlana Kiritchenko. 2016. Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Berlin, Germany. Association for Computational Linguistics.

Swapna Somasundaran and Janyce Wiebe. 2009. Recognizing Stances in Online Debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, pages 226–234, Stroudsburg, PA, USA. Association for Computational Linguistics. Event-place: Suntec, Singapore.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124, Los Angeles, CA. Association for Computational Linguistics.

Dhanya Sridhar, Lise Getoor, and Marilyn Walker. 2014. Collective stance classification of posts in online debate forums. In *Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pages 109–117, Baltimore, Maryland. Association for Computational Linguistics.

Qingying Sun, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. 2018. Stance detection with hierarchical attention network. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2399–2409, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 327–335, Sydney, Australia. Association for Computational Linguistics.

Orith Toledo-Ronen, Roy Bar-Haim, Alon Halfon, Charles Jochim, Amir Menczel, Ranit Aharonov, and Noam Slonim. 2018. Learning sentiment composition from sentiment lexicons. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2230–2241, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Douglas Walton, Christopher Reed, and Fabrizio Macagno. 2008. *Argumentation Schemes*. Cambridge University Press.

Rui Wang, Deyu Zhou, Mingmin Jiang, Si Jiasheng, and Yang Yang. 2019. A survey on opinion mining: from stance to product aspect. *IEEE Access*, PP:1–1.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics. Event-place: Vancouver, British Columbia, Canada.