

Constructing A Dataset of Support and Attack Relations in Legal Arguments in Court Judgements using Linguistic Rules

Basit Ali, Sachin Pawar, Girish K. Palshikar, Rituraj Singh

TCS Research

Tata Consultancy Services, Pune, India.

{ali.basit, sachin7.p, gk.palshikar, singh.rituraj}@tcs.com

Abstract

Argumentation mining is a growing area of research and has several interesting practical applications of mining legal arguments. Support and Attack relations are the backbone of any legal argument. However, there is no publicly available dataset of these relations in the context of legal arguments expressed in court judgements. In this paper, we focus on automatically constructing such a dataset of Support and Attack relations between sentences in a court judgment with reasonable accuracy. We propose three sets of rules based on linguistic knowledge and distant supervision to identify such relations from Indian Supreme Court judgments. The first rule set is based on multiple discourse connectors, the second rule set is based on common semantic structures between argumentative sentences in a close neighbourhood, and the third rule set uses the information about the source of the argument. We also explore a BERT-based sentence pair classification model which is trained on this dataset. We release the dataset of 20506 sentence pairs – 10746 Support (precision 77.3%) and 9760 Attack (precision 65.8%). We believe that this dataset and the ideas explored in designing the linguistic rules and will boost the argumentation mining research for legal arguments.

1. Introduction

A court judgement document describes various details about a particular court case such as names of the appellants and the respondents, facts related to the case, arguments by the lawyers, witness testimonies, evidences, statutes, relevant prior cases, decision by the judges, and rationale behind the decision. Sentences corresponding such details are often related to each other with certain semantic relations. Identifying these relations among the sentences in a court judgment, is important for better understanding and representation of the court judgement, especially the legal arguments mentioned in the judgement. This is useful for several applications such as identifying legal arguments of each party in a case, suggesting appropriate arguments in a certain scenario, prior case retrieval, prediction of the court decision, etc.

In this paper, we focus on two key relation types between sentences – Support and Attack. We define these relations as follows – for sentences P and C , the relation P Support C (P Attack C) holds if:

1. C is a proposition (fact or opinion or belief) which can either be true or false AND P strengthens (or weakens) the truth level / truth possibility of C ; or
2. C is an action AND P justifies (or denies) or explains (or opposes) C ; or
3. C is an event AND P strengthens (or weakens) the possibility that C actually happened.

Here, P is called the *premise* and C is called the *claim*. In this paper, we assume P and C are two separate sentences, although it is possible that they both are parts of the same sentence. There is no restriction on the

kind of sentence P is required to be in this relation. Table 1 shows examples of Support and Attack relations observed between sentences in court judgements.

Support and Attack relations observed in the legal domain in court judgement documents, have different characteristics as compared to these relations observed in other domains like *debates*. Often in Argument Mining literature, the *claim* sentence can only be a proposition which can be true or false (Palau and Moens, 2009), whereas in legal domain, the *claim* sentence in Support or Attack relation can correspond to an action or event (e.g., The compensation for tree growth is accordingly **enhanced**... in example 5 in Table 1). In legal domain, in addition to supporting facts or evidences, another types of support (or attack) we observe are – (i) from prior cases (example 2 in Table 1), or (ii) from interpretation of legal terms/provisions as they apply to a given court case (example 3 in Table 1). A sentence can be a *premise* and a *claim* simultaneously where it is being supported (or attacked) by one sentence and it itself is supporting (or attacking) another sentence (Poudyal et al., 2020) (examples 6 and 7 in Table 1).

Given the characteristics of the Support and Attack relations in legal domain, it is quite challenging to automatically identify them. One key challenge is that the premise and claim sentences can be of various types such as facts (P in example 1, C in example 5), opinions (P in example 2, C in example 4), events (P in example 10), or actions (P and C in example 8). Another challenge is that the Support and Attack relations may be expressed explicitly or implicitly. For example, in example 1, the Attack relation is explicitly observed because of direct contradiction between P (on the basis of five criteria) and C (no objective

	Relation	Premise (P)	Claim (C)
1	Attack	The selectors were asked to interview candidates on the basis of the five criteria prescribed to which we have made reference earlier.	It was next urged that no objective criterion was fixed for interview.
2	Attack	Neither the judgment of Privy Council nor the judgment of this Court from which support was sought for the argument, can furnish such support.	He sought to obtain support for that argument from the judgment of this Court in Bishan Singh v. Khazan Singh and the judgment of the Privy Council in Deonandan Prashad Singh v. Ramdhari Chowdhri.
3	Attack	Though the language of s. 2 might, in the abstract, be susceptible of the construction which the petitioners seek to put upon it, in the context that is not, in our opinion, its true meaning.	According to the petitioners, "enactment in force" in s. 2 must be construed as meaning provisions of a statute which are valid and enforceable.
4	Support	But the revision of rent can not be insisted on by the landlord as a condition precedent to re-entry by the tenant.	Therefore, the landlord in the present case was not justified in offering the premises to the tenants for re-entry by qualifying the offer for payment of a higher rate of rent.
5	Support	Having regard to the large number of trees and to the fact that some of them were timber trees, it would be appropriate to award a lump-sum of Rs. 7,500 under this head.	The compensation for tree growth is accordingly enhanced from Rs. 355.85 to Rs. 7,500.
6	Support	In the present case, there is nothing to show that the policemen were making false statements in the court.	We can not assume that every statement of a policeman is necessarily false.
7	Support	The policemen had no enmity with the accused.	In the present case, there is nothing to show that the policemen were making false statements in the court.
8	Support	These cheques were presented by the appellant for encashment, but the same were returned with the endorsement Payment stopped by the drawer.	The appellant, therefore, served notices on the respondent, calling upon him to pay the amount of cheques within 15 days of the receipt of the notice.
9	Support	In particular, the denial of access to a lawyer, doctor, relative or friend and the absence of any realistic possibility of being brought before a court to test the legality of the detention meant that he was left completely at the mercy of those holding him.	The Court considers that in this case insufficient safeguards were available to the applicant.
10	Support	His calculations went wrong and he failed in his attempt to avoid the accident.	Clearly, therefore, the accident occurred not on account of his negligence but due to an error of judgement in the circumstances of the situation.
11	Attack	Even if we proceed on the basis that to practice as an advocate is a fundamental right, no right to be allotted a chamber within the Court premises follows from it.	To make a chamber available to him is an integral part of his guaranteed fundamental right.

Table 1: Examples of relations between sentences

criterion was fixed). On the other hand, in example 7, P indirectly supports C and the background common sense knowledge (if X has enmity towards Y then he is more likely to give false statement to implicate Y) is not mentioned explicitly.

In this paper, we explore three sets of rules to automatically label sentence pairs as Support or Attack. The first set of rules exploits multiple discourse connectors connecting consecutive sentences. The second set of rules focuses on finding sentence pairs in a close neighbourhood which share common *semantic structures* (defined later in detail) and at least one of the sentence is argumentative in nature. The third set of rules employs a kind of *distant supervision* that is based on knowledge of the *source* of arguments in a court judgement. Here, source of an argument is the party who is making that argument, e.g., appellant or respondent. The distant supervision is based on the intuition that an appellant’s argument is more likely to *attack* a respondent’s argument whereas it is more likely to *support* appellant’s another argument. Using these sets of rules, we automatically create a large labeled dataset¹ and the precision of these labels is estimated by manually verifying a random subset of this dataset.

The paper is organized as follows. Section 2 describes the relevant past work. Section 3 describes the proposed three sets of rules in detail along with multiple examples. Section 4 provides certain key statistics regarding the dataset created by the proposed rules. Section 5 explores the performance of a BERT-based sentence pair classification model on our dataset. Finally, we conclude in Section 6 and also identify some potential future directions.

¹The dataset would be shared upon request.

2. Related Work

Legal argumentation dataset: Recently, Poudyal et al. (Poudyal et al., 2020) released a dataset in legal domain where arguments are annotated in ECHR (European Court of Human Rights) corpus. An argument is identified by grouping together corresponding *premise* and *conclusion* sentences. Here, we observed that the distinction between Support and Attack relation was not explicitly identified. Example 9 in Table 1 is from this ECHR corpus and expresses the Support relation.

Other argumentation datasets: There are several corpora related to argumentation where argument is represented in different structure such as AML (Argument Markup Language) (Reed et al., 2008), AIFdb (Lawrence et al., 2012) which is an open database containing argumentation structures in argument interchange format (AIF), and Internet Argument Corpus (IAC) (Walker et al., 2012) for political debates where quotes response pairs are extracted. Apart from different argument structures corpora, various online tools are available where users can lookup an argument in the search engine to support users in finding arguments and forming opinions on controversial topics². Another similar system is Argument Text (Stab et al., 2018) which extracts sentential arguments from large sum of arbitrary text and another is online visualization tool (OVA) (REED, 2014) used to analyze and annotate the argumentative structure of natural language text. Some tools like ArguminSci³ are used to analyze the scientific publication by identifying argumentative components. Although there are several general domain datasets for arguments like persuasive essays (Stab and Gurevych, 2017), political

²<http://www.argumentsearch.com/>

³<https://github.com/anlausch/ArguminSci>

speeches (Menini et al., 2018) or debates, there is no publicly available dataset of Support and Attack relations in the context of legal arguments expressed in court judgements. Cocarascu and Toni (2017) used a dataset for predicting Support and Attack relations between sentences which covered various domains such as movies, technology, and politics.

Datasets based on Indian Supreme Court corpus:

Using the corpus of Indian Supreme Court judgements, some other datasets have been prepared in the past. For example, Bhattacharya et al. (2019b) released a dataset of 50 court judgements where the sentences are annotated with rhetorical roles such as facts, rulings by lower court, argument, statute, precedent, etc. This corpus is also used for several other tasks such as prior case retrieval (Bhattacharya et al., 2019a; Ghosh et al., 2020; Ali et al., 2021), document summarization (Bhattacharya et al., 2021), identifying textual similarity between legal court case reports (Mandal et al., 2021), and court judgment prediction (Malik et al., 2021).

Using weak supervision for training data creation:

Recent deep learning models need very large annotated datasets for training. Manually annotating a large number of instances to create such training datasets, is very costly in terms of time, efforts, and money. Hence, there have been attempts to use weak supervision or distant supervision (Mintz et al., 2009) to automatically create annotated datasets which can be used to train machine learning models. The Snorkel framework (Ratner et al., 2017) has been developed for the very same purpose of creating labeled training data automatically and rapidly. Snorkel enables writing multiple *labeling functions* (LFs) where each LF expresses an arbitrary heuristic for predicting certain class labels. These LFs can have unknown accuracies and correlations but Snorkel denoises their outputs and combines their predictions to arrive at a final probability distribution over labels for each instance. A large training set can then be constructed rapidly using these automatically assigned *soft* labels. Here, *soft* labels are probability distribution over labels. The Snorkel framework has been used for providing weak supervision in various tasks such as relation extraction (Mallory et al., 2020) and sentiment analysis (Jain, 2021). On the similar lines, our aim is to automatically create a dataset of sentence pairs labeled with Support and Attack relations, using a set of rules. Our rules can easily be adapted as labeling functions in the Snorkel framework.

3. Rules for Dataset Construction

In this section, we describe our proposed rules for identifying whether any Support or Attack relation exists for a sentence pair. The rules are categorized into three different sets where each set of rules focuses on a different linguistic characteristic.

3.1. Problem Definition

For each set of rules, the input and output details are as follows:

Input: A court judgment document $D = [s_1, s_2, \dots, s_n]$ containing sequence of n sentences.

Output: (i) Sentence pairs $\langle s_i, s_j \rangle$ such that s_i supports s_j , and (ii) Sentence pairs $\langle s_i, s_j \rangle$ such that s_i attacks s_j .

3.2. R1: Linguistic Rules based on Discourse Connectors

This set of rules exploits multiple discourse connectors and the dependency tree structures of sentences. It is motivated by the observation by Eckle-Kohler et al. (2015) that discourse markers are very useful for discriminating claims and premises in argumentative discourse. The Basic idea behind these rules is to find certain discourse connectors which connect two sentences such that the two sentences are *related* and they have a Support or Attack relation between them. The discourse connectors can be single words (e.g., *therefore, however*), phrases (e.g., *it follows that, on the other hand*), or complete sentences (e.g., *We do not accept this argument.*). Any two sentences are said to be *related* if they satisfy certain *overlap* conditions.

Overlap conditions: Any sentence pair for which the rules identify Support or Attack relation, should not be completely unrelated. The overlap conditions ensure that there is an overlap of at least K content words (nouns, verbs, or adjectives excluding stop words) between the two sentences (we use $K = 2$). Two words are considered to be overlapping if they are exactly same, have exactly same root form (e.g., *terminate* \Leftrightarrow *terminated*), are part of a single synset in WordNet (synonyms, e.g., *building* \Leftrightarrow *construction*), or have high cosine similarity (we used 0.8 as a threshold) using GloVe (Pennington et al., 2014) word embeddings (e.g., *witness* \Leftrightarrow *testimony*, *cosine_sim* = 0.8).

The rules to identify Support or Attack relations are explained in detail below and the example sentence pairs identified by each rule are shown in Table 2.

Rule using causal discourse connectors: If there exists a verb v in sentence s_i such that:

- there exists a causal discourse connector modifying v which is:

- a word dw_{causal} modifying v with dependency relation $advmod$ (adverbial modifier), OR

- a phrase dp_{causal} modifying v with dependency relations $prep$ (prepositional modifier) or $ccomp$ (clausal complement)

- there is no other main verb in s_i left of v

- sentence s_{i-1} contains at least one *complete* clause (i.e., it should have at least one *verb* with *subject* as well as *object* or *prepositional object*)

- the *overlap conditions* (described above) hold between s_{i-1} and s_i

THEN s_{i-1} Supports s_i

Here, $dw_{causal} \in \{\text{therefore, accordingly, thus, consequently, hence, thereby}\}$ and

$dp_{causal} \in \{\text{follows that, it followed, in conclusion}\}$.

Rule using *contrast-indicating discourse connectors*:

If there exists a verb v in sentence s_i such that:

- there exists a contrast-indicating discourse connector modifying v which is:

- a word $dw_{contrast}$ modifying v with dependency relation *advmod* (adverbial modifier), OR

- a phrase $dp_{contrast}$ modifying v with dependency relations *prep* (prepositional modifier) or *mark* (subordinate clause)

- there is no other main verb in s_i left of v

- sentence s_{i-1} contains at least one *complete* clause

- the *overlap conditions* hold between s_{i-1} and s_i

THEN s_i Attacks s_{i-1}

Here, $dw_{contrast} \in \{\text{however, conversely}\}$ and $dp_{contrast} \in \{\text{on the other hand, on the contrary, although}\}$.

Flip sentence rule: Here, we refer to a sentence as a *flip* sentence, if it clearly indicates that the upcoming sentences are going to *attack* the arguments described just before that sentence. For example, We are unable to accept this argument. Such sentences are often made by the judges before they start providing rationale of their decision. We identify such sentences using a regular expression pattern. If s_i is a *flip* sentence such that:

- the *overlap conditions* hold between s_{i-1} and s_{i+1}

THEN s_{i+1} Attacks s_{i-1}

Rule using an explicit support-indicating phrase:

Here, we want to make use of phrases which explicitly indicate support to the argument or submission mentioned in the previous sentence, e.g., in support of this submission. If there exists a support indicating phrase $p_{support}$ in s_i such that:

- a argumentation event noun (e.g., *submission, contention, allegation*) is the direct or indirect child of the word *support* in the dependency tree

- a marker such as *this, aforesaid, above* which modifies the argumentation event noun and links it to the previous sentence

- the *overlap conditions* hold between s_i and s_{i-1}

THEN s_i Supports s_{i-1} .

3.3. R2: Linguistic Rules based on Semantic Structure Overlap

We explored another set of linguistic rules which are based on high semantic similarity between two nearby sentences in a court judgement document. We capture the semantic similarity using *common semantic structures* shared between any two sentences s_1 and s_2 . A *common semantic structure* consists of a *head* (noun or verb) which is present in both the sentences. The head may not correspond to an exactly matching word pair in the two sentences, but any two words $w_1 \in s_1$ and $w_2 \in s_2$ can correspond to the head such that w_1 and w_2 satisfy the overlapping conditions described earlier (same root forms, WordNet synonyms, or high

cosine similarity of GloVe embeddings) or they may be WordNet antonyms. Additionally, the common semantic structure also consists of *arguments* of the head. Each argument in the common semantic structure corresponds to a word pair in the two sentences satisfying the overlapping conditions. Here, an argument is – i) a descendant of the head in the dependency tree if the head is a verb, or ii) a descendant of the head’s lowest verb ancestor if the head is a noun (e.g., in s_1 of the first sentence pair in Table 3, the lowest verb ancestor of the head *conviction* is *stand*). Each argument has to satisfy this condition in both the sentences. Table 3 shows two examples of sentence pairs and their common semantic structures.

We consider only *event-indicating nouns* as a potential head of a common semantic structure. A noun is an event-indicating noun if *event.n.01* is its ancestor in WordNet hypernym tree for any of its top 2 senses. For example, *possession* is an event-indicating noun because hypernyms for its first sense in WordNet are: *possession.n.01* → *control.n.05* → *activity.n.01* → *act.n.02* → *event.n.01*. Examples of nouns which are not event-indicating nouns are – *appellant, property, lawyer*.

Here, the intuition is that if two sentences are appearing close to each other (we consider a window of 3 sentences) and have common semantic structures then it is very likely that there is a Support or Attack relation between them, provided some additional conditions are satisfied. One reason behind this assumption is that it is very unlikely that a certain fact or argument is repeated in multiple sentences in a court judgement document and hence if such common semantic structures are observed across two different sentences, then the sentences are more likely to have Support or Attack relation rather than Entailment relation. Algorithm 1 describes the rule in detail. The subroutine *is_argumentative* checks whether any sentence is argumentative in nature or not. It uses a classifier trained using combined training data from multiple sources. The positive examples (9555) are collated from ECHR corpus (Poudyal et al., 2020) (2694), persuasive essays corpus (Stab and Gurevych, 2017) (6089) and Argument sentences from rhetorical roles corpus (Bhattacharya et al., 2019b) (772). The negative examples (8033) are randomly sampled from Indian Supreme Court corpus (6000) excluding sentences containing argumentative keywords (e.g., *contended, concluded, hence*) and Fact sentences from rhetorical roles corpus (Bhattacharya et al., 2019b) (2033). The classifier used is a BERT-based sentence classifier which combines the [CLS] representation of a sentence and attention weighted average of the other tokens to get the overall representation of the sentence.

Algorithm 1 first finds the common semantic structures between two nearby sentences. The subroutine *AreOpposite* is used to check if the common semantic structure is opposite in meaning in the two sentences.

Rule using causal discourse connectors (s_{i-1} Support s_i):	
s_{i-1}	He urged that upon a true interpretation of the provisions of the KST Act, particularly Sections 13 and 15 read with Section 2(f-2), it would be clear that the First Respondent was the entity to whom the business of the Defaulting Company had been transferred.
s_i	Therefore , Mr. Hegde urged, the sales tax dues of the Defaulting Company could rightfully be claimed and recovered from the First Respondent.
Rule using contrast-indicating discourse connectors (s_i Attack s_{i-1}):	
s_{i-1}	On appeal the Appellate Authority held that the shops and chobaras were in good condition and that the landlord was not, in good faith, wanting to replace the building, when he had no means to build it.
s_i	The High Court, however , allowed the revision petition filed before it holding that upon the evidence on record it had been established beyond doubt that the landlord genuinely and bona fide required the premises for re-building.
Flip sentence rule (s_{i+1} Attack s_{i-1}):	
s_{i-1}	Mr. Gokhale, however, argued that it was no part of the appellant 's duty to produce the accounts unless he was called upon to do so and the onus was upon the respondents to prove the case and to show that the Dargah was the owner of plot No. 134.
s_i	We are unable to accept this argument as correct.
s_{i+1}	Even if the burden of proof does not lie on a party the Court may draw an adverse inference if he withholds important documents in his possession which can throw light on the facts at issue.
Rule using an explicit support-indicating phrase (s_i Support s_{i-1}):	
s_{i-1}	Ultimately the tribunal came to the conclusion that in the circumstances of the case it would be fair to allow the appellant about Rs. 165 to 170 lakhs as annual provision for the said items.
s_i	In support of this conclusion the tribunal has relied on the fact that for the two years 1952-53 and 1953-54 the appellant had spent about Rs. 339.76 lakhs for the purpose of rehabilitation, replacement land modernisation and that works at the average of Rs. 170 lakhs per year.

Table 2: Examples of sentence pairs labeled by the rules based on discourse connectors

Sentence pair	Common Semantic Structure
s_1 : The conviction of an accused, or the finding of the Court that he is guilty, does not stand washed away because that is the sine-qua-non for the order of release on probation.	Head: conviction \Leftrightarrow sentence (identified as synonyms as per WordNet)
s_2 : The order of release on probation is merely in substitution of the sentence to be imposed by the Court.	Arguments: order, probation, court, release.
s_1 : There was no commission of the offense of undue influence by anybody with the connivance of the respondent and the result of the election was not materially affected.	Head: influence (negated) \Leftrightarrow influence (exact match)
s_2 : Some of the allegations made in paragraphs 8(3) and (13) of the petition would be sufficient pleading of commission of undue influence under Section 18(1)(a).	Arguments: commission

Table 3: Examples of common semantic structure between sentence pairs

If the common semantic structure is opposite in meaning in the sentences then Attack relation is identified, otherwise Support relation is identified. We now describe how the subroutine *AreOpposite* works. If the matching words (head or its arguments) are opposite in the meaning, either by WordNet-based antonyms or any word satisfying *negation rules* (described later) then the common semantic structure is said to have opposite meaning in the two sentences. For example, guilt in $s_i \Leftrightarrow$ innocent in s_j ; imply in $s_i \Leftrightarrow$ mean in s_j where mean is satisfying negation rules. The negation rules based on dependency tree are as follows:

- If a word (verb or noun) has a child with *neg*

dependency relation (e.g., no, not), then it is negated.

- If a verb v has a negative verb v' as a parent with *xcomp* relation (or a negative adjective with *acompl* relation)⁴ (e.g. failed, incompetent), then v is negated (e.g., the Act which has failed to **comply**.)

- If a verb v has an auxiliary verb as a parent with *xcomp* dependency relation and the auxiliary verb has a child with *neg* dependency relation, then v is negated. However, if the auxiliary verb also has *acompl* dependency relation with some negative adjectives (e.g., difficult, challenge, or hard), then v

⁴<https://ptrckprry.com/course/ssd/data/negative-words.txt>

should not be negated (e.g., hold will not be negated in: It is not difficult to **hold** that the allegation...).

- If not only.*?VERB pattern is present then the verb v should not be negated (e.g., The apprehension must not only be **entertained** but must appear to the court to be reasonable.)
- If a noun n or verb v has without as parent with *pobj* or *pcomp* dependency relation respectively then n or v is negated (e.g., without a **record**; without **recording**)
- If a noun n has a negative word as a parent, then n is negated (e.g., failure of **justice**).
- If nothing, nobody or none is a child of a verb v with *nsubj* dependency relation, then v is negated and if v is connected to another verb through dependency relations *xcomp* or *conj* then the negation is transferred to v' also.

Table 3 shows the examples of sentence pairs for which Algorithm 1 identifies Support (for the first pair) and Attack (for the second pair) relation. For both the pairs, both s_1 and s_2 are identified as “argumentative” and there also exists a common semantic structure as shown in Table 3. However, the subroutine *AreOpposite* identifies that the common semantic structure for the second pair has opposite meaning in the two sentences and hence Attack relation is identified.

input : A court judgment document $D = [s_1, s_2, \dots, s_n]$ containing sequence of n sentences

output: Set of triplets of the form $\langle s_i, R, s_j \rangle$ where $R \in \{\text{Support, Attack}\}$

$O := \{\}$;

```

foreach sentence pair  $\langle s_i, s_j \rangle \in D$  s.t.  $j - i \leq 3$  do
  if is_argumentative( $s_i$ ) OR is_argumentative( $s_j$ )
  then
     $CSS_{ij} :=$  common semantic structures in  $s_i$  and  $s_j$ ;
    if  $|CSS_{ij}| \geq 2$  then
       $css_{ij}^* :=$  common semantic structure  $\in CSS_{ij}$ 
      having the head words closest to the roots ;
      if AreOpposite( $css_{ij}^*, s_i, s_j$ ) then
         $O := O \cup \langle s_j, \text{Attack}, s_i \rangle$ ;
      else
         $O := O \cup \langle s_i, \text{Support}, s_j \rangle$ ;

```

return O ;

Algorithm 1: Rule using common semantic structures

3.4. R3: Distant Supervision using Argument Source

The third set of rules relies on the the party who is making a certain argument. We initially identify sentences containing arguments by contesting parties as well as lower courts mentioned in the court judgement documents. To identify these sentences, we follow the following steps for each sentence s (the subroutine *identify_argument_and_source* in Algorithm 2):

- s should contain at least one high confidence

argument-indicating keyword (e.g., argued, contended, argument, contention)

- The argument-indicating keyword should have the word that as its descendant in the dependency tree of s . This indicates that s contains what the actual argument was as a clausal complement.
- To the left of the word that, s should contain argument source indicating keyword (e.g., appellant, respondent, High Court)
- Depending on which argument source indicating keyword is present in s , it is categorized in one of the three types – A (argument by Appellant), R (argument by Respondent), and L (argument by Lower court).

Table 4 shows example sentences identified by the above rules, along with their labels indicating the argument sources. Intuitively, the arguments by an appellant would attack the arguments by a respondent and vice versa. The arguments by an appellant would also attack arguments by the lower court because it is the appellant who has brought the matter to the Supreme Court after unfavourable decision in the lower court. Also, the arguments by appellant (or respondent) would support each other. Algorithm 2 describes the rule in detail. Consider the following sentence pair for which Algorithm 2 identifies the Attack relation.

S_1 : The appellants contended that the High Court committed an error in relying on Ex. A2 as it was nearly 3 years prior to the acquisition and there was a steep increase in the value of land during that period.

S_2 : The respondent also contended that the increase in value per year was rightly taken as 10% and that being the standard increase, should not be interfered with.

S_1 is identified as type A and S_2 is identified as type R. The common semantic structure is – Head: increase \Leftrightarrow increase (exact match); Argument: value.

4. Dataset

We use the corpus of Indian Supreme Court judgements from 1952 to 2012 which are available at <http://liiofindia.org/in/cases/cen/INSC/>. This corpus consists of 30034 documents containing 4062500 sentences and 126706200 words. The average document length is 135 sentences with the standard deviation of 190 and the median document length is 99 sentences. The average sentence length is 31 words with a standard deviation of 24 and the median sentence length is 24 words. As our aim is to create a dataset of sentence pairs, the total possible candidate intra-document sentence pairs in this corpus are around 547 million, out of which only a small fraction of sentence pairs (20506) were identified by our rules.

We apply the rules described in Section 3 to identify the sentence pairs with Support or Attack relations. Table 5 shows the dataset details in terms of coverage and estimated precision of each rule. The dataset consists

Example sentence	Argument source
The appellant contested the election petitions, contending <u>that</u> there was no improper acceptance of nomination paper of the candidate in question and that the appellant 's election was not materially affected.	A
Mr. K. Parasaran, appearing on behalf of the respondent in Civil Appeal arising out of S.L.P.(c) No. 13401/2003 submitted <u>that</u> the objection taken as to the jurisdiction of the Patna High Court to issue a writ against the State of Jharkhand was misconceived.	R
The High Court said <u>that</u> the provisions contained in section 108 of the Act are directory because non-compliance with section 108 of the Act is not declared an offence.	L

Table 4: Examples of sentences labeled as A (argument by Appellant), R (argument by Respondent), and L (argument by Lower court)

Rule	Support		Attack		Total	
	Cov.	Prec.	Cov.	Prec.	Cov.	Prec.
R1 (Discourse Connectors)	3619	0.853	4008	0.840	7627	0.847
R2 (Semantic Structure Overlap)	5137	0.787	3516	0.400	8653	0.593
R3 (Distant Supervision)	1990	0.680	2236	0.733	4226	0.707
All Rules	10746	0.773	9760	0.658	20506	0.716
R1 (all)+R2 (only Support)+R3 (only Attack)	8756	0.820	6244	0.787	15000	0.803

Table 5: Coverage and estimated precision of each set of rules across Support and Attack relations

input : A court judgment document $D = [s_1, s_2, \dots, s_n]$ containing sequence of n sentences
output: Set of triplets of the form $\langle s_i, R, s_j \rangle$ where $R \in \{\text{Support, Attack}\}$
 $O := \{\}$; $S_A, S_R, S_L := \{\}$;
foreach $s_i \in D$ **do**
 $label := identify_argument_and_source(s_i)$;
 if $label = "A"$ **then** $S_A := S_A \cup \{s_i\}$;
 else if $label = "R"$ **then** $S_R := S_R \cup \{s_i\}$;
 else if $label = "L"$ **then** $S_L := S_L \cup \{s_i\}$;
foreach sentence pair $\langle s_i, s_j \rangle \in D$ s.t. $i < j$ and $s_i, s_j \in S_A \cup S_R \cup S_L$ **do**
 $CSS_{ij} :=$ common semantic structures in s_i and s_j ;
 if $|CSS_{ij}| = 0$ **then** **continue**;
 if $(s_i \in S_A \text{ AND } s_j \in S_R)$ **OR** $(s_i \in S_R \text{ AND } s_j \in S_A)$ **OR** $(s_i \in S_L \text{ AND } s_j \in S_A)$ **then**
 | $O := O \cup \langle s_j, \text{Attack}, s_i \rangle$;
 else if $(s_j \in S_L \text{ AND } s_i \in S_A)$ **then**
 | $O := O \cup \langle s_i, \text{Attack}, s_j \rangle$;
 else if $(s_i \in S_A \text{ AND } s_j \in S_A)$ **OR** $(s_i \in S_R \text{ AND } s_j \in S_R)$ **then**
 | $O := O \cup \langle s_i, \text{Support}, s_j \rangle$;
return O ;

Algorithm 2: Distant supervision based rule

of 20506 sentence pairs where the average and median sentence length is 44 with the standard deviation of 16. For estimating precision, we randomly selected 450 sentence pairs where 150 pairs were selected from each set of rules, equally distributed among Support and Attack. Three co-authors of this paper manually verified each of these sentence pairs labeled with Support or Attack and marked whether the label is correct or incorrect. We have provided a few examples of incorrect labels in the Appendix. Moreover, the Cohen's

Kappa coefficient for the inter annotator agreement between two of the authors is 0.68 on 150 sentence pairs (selected from each set of rules, 50 each). This subset of 450 manually verified sentence pairs were used to estimate approximate precision of each rule.

Analysis: The overall coverage of the dataset is 20506 sentence pairs which are labeled with approximately 71.6% precision. It was observed that the first set of rules based on discourse connectors has the highest precision of 84.7% with the coverage of 7627 sentence pairs. The second set of rules based on common semantic structure resulted in poor precision of 59.3%. Although, the precision of Support sentence pairs identified by this set of rules is much higher at 78.7%. The poor precision is mainly the result of Attack sentence pairs which are identified with just 40% precision. Here, we observed that most of the errors in these Attack sentence pairs were actually Support examples but due to incorrect capturing of *negation*, the rules incorrectly predicted them as Attack. Thus, identifying *negation* reliably using rules is very challenging, and we wish to pursue this further as our future work. The third set of rules based on distant supervision has the precision of 70.7%. However the precision of Support sentence pairs is merely 68% compared to the 78.7% of the second set of rule (Support) whereas for the precision of Attack sentence pairs is 73.3%. Hence, we observed that the combination of all sentence pairs labeled by R1, only Support pairs labeled by R2, and only Attack pairs labeled by R3 gives us a subset of 15000 sentence pairs with a precision of 80.3%.

5. Supervised Sentence Pair Classifier

We explored how the BERT-based (Devlin et al., 2018) supervised sentence pair classification model performs

on our dataset constructed using the proposed rules. It follows the standard practice for sentence-pair classification tasks like predicting textual entailment (Devlin et al., 2018) or a binary classification task to decide that whether two sentences are part of the same argument or not (Poudyal et al., 2020). All the 20506 sentence pairs labeled with Support and Attack are used as positive examples for classification. We generate 20000 negative sentence pairs (NO_REL, i.e., having neither Support nor Attack relation between them) consisting of 4000 pairs from each of the following types:

- N_1 : Randomly selected intra-document sentence pairs from the whole corpus
- N_2 : Sentence pairs where one of the sentence contains a causal or contrast-indicating discourse marker and another sentence is selected randomly
- N_3 : Sentence pairs where one of the sentence is identified as A, R, or L (by the third set of rules R3) whereas another sentence is selected randomly
- N_4 : Sentence pairs where there is at least one common semantic structure but the two sentences are from two different court judgement documents
- N_5 : Sentence pairs randomly selected from the first set of rules (R1) but the order of sentences is flipped

Hence, the overall classification problem is to predict one of the three labels for any given sentence pair – Support, Attack, and NO_REL. The architecture of the classification model is as follows. We use the pre-trained BERT model where the two sentences are provided as input separated by the special token [SEP] and the special token [CLS] as the first token. The BERT encoding output (768-dim) for the [CLS] token is considered as the aggregated representation of the sentence pair which is fed into a hidden layer (384 units) and further to 3 output units corresponding to the 3 labels. Softmax function is used at the output units to get the probability distribution over the 3 labels. We carried out 5-fold cross validation over our dataset and Table 6 shows the evaluation results. Overall accuracy of classification is 84% whereas F1 scores for Support and Attack prediction are 0.792 and 0.759, respectively. We also computed precision for each type of negative sentence pairs – N_1 : 0.969, N_2 : 0.931, N_3 : 0.936, N_4 : 0.971, and N_5 : 0.967.

Implementation details: The following hyperparameters are used – batch size of 64, learning rate of 0.0001 with Adam optimizer, 5 epochs. We applied the dropout layer before and after the hidden layer with a probability of 0.1 and 0.4 respectively, to avoid overfitting. Cross-entropy was used as the loss function. Moreover, we also fine-tuned the last two encoder layers of BERT while training on our dataset.

Sentence similarity scores: We also analyzed how sentence similarity scores are distributed among the positive (Support and Attack), the negative (NO_REL), and random sentence pairs. We computed the sentence similarity between the two sentences in a sentence pair in two ways – i) word-level Jaccard simi-

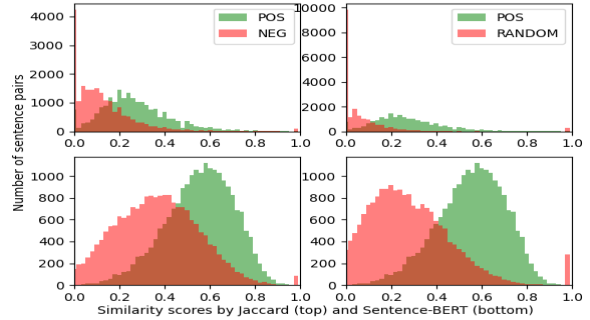


Figure 1: Histogram of the sentence similarity scores – positive vs. negative (left) and positive vs. random sentence pairs (right)

Relation	Precision	Recall	F1	Accuracy
Support	0.778	0.806	0.792	-
Attack	0.746	0.772	0.759	-
Overall	-	-	-	0.840

Table 6: 5-fold cross-validation results over our dataset by the BERT-based sentence pair classification model

ilarity between the two sentences, considering the overlap conditions mentioned in Section 3.2, and ii) cosine similarity between sentence representations obtained by the Sentence-BERT model (Reimers and Gurevych, 2019). Figure 1 shows the histogram of these sentence similarity scores across the positive and negative sentence pairs as well as across the positive and the random sentence pairs.

6. Conclusions and Future Work

In this paper, we release a dataset of 20506 sentence pairs from court judgements that are labeled with Support or Attack relations. To the best of our knowledge, this is the first such dataset for legal arguments expressed in court judgement documents. The dataset is created automatically by using three sets of rules based on linguistic knowledge and distant supervision. We estimated the precision of the automatically assigned labels by manually verifying a randomly sampled sentence pairs from the dataset. The overall precision of the dataset was found to be 71.6% but we also identified a smaller subset of the dataset (15000 sentence pairs) which has a much higher precision of 80.3%. We also explored a BERT-based sentence pair classifier on this dataset and observed the F1-scores of 79.2% and 75.9%, respectively for Support and Attack with 5-fold cross-validation. In future, we plan to improve the coverage and the precision of the existing rules further and explore more sophisticated model architectures for the supervised sentence pair classification problem. Also, we wish to extend the existing rules so that Support and Attack relations are identified at a more fine-grained level, i.e., identifying a relation between sentence clauses instead of between two sentences.

Rule	Sentence Pair	Assigned	True
R1	<p>S1: In my judgment that is not contemplated by the power conferred to reserve which can only mean for the future.</p> <p>S2: As this point however has not been argued I do not desire to rest my judgment on it, but have mentioned it to draw attention to another feature of the notification which deserves consideration.</p>	Attack	NO_REL
R2	<p>S1: For the reasons stated above and in view of the conduct of the <u>Advocate</u> seen in the light of the surrounding circumstances we are clearly of opinion that the Advocate should, by reason of his having indulged in conduct unworthy of a member of the honourable profession to which he belongs, be suspended from <u>practice</u> for some time.</p> <p>S2: He is an <u>Advocate</u> of this Court and according to a majority decision of this Court he is entitled, under the Supreme Court Advocates (<u>Practice</u> in High Courts) Act, to exercise his profession in all Courts throughout the Union of India.</p>	Support	Attack
R3	<p>S1: The petitioners contend that allowing their competitors to construct such a working platform and disallowing construction of the platform in the case of the Express Newspapers Pvt. Ltd. is clearly violative of the petitioners' fundamental right to equality before the law guaranteed by Article 14 of the Constitution.</p> <p>S2: The petitioners contend that the slab of the working platform constructed by them does not fall within the meaning of the expression 'covered area' in sub-cl.(22) of cl.2 of the Building Bye-laws, since it is below the plinth level.</p>	Support	NO_REL

Table 7: Examples of error cases by the proposed rules

Appendix

Table 7 shows some examples of sentence pairs for which rules have identified incorrect labels. For each sentence pair, the label assigned by rules and the true label is shown.

7. References

- Ali, B., More, R., Pawar, S., and Palshikar, G. K. (2021). Prior case retrieval using evidence extraction from court judgements. In *Automated Semantic Analysis of Information in Legal Text (ASAIL) workshop at 18th International Conference on Artificial Intelligence and Law (ICAIL)*.
- Bhattacharya, P., Ghosh, K., Ghosh, S., Pal, A., Mehta, P., Bhattacharya, A., and Majumder, P. (2019a). Fire 2019 aila track: Artificial intelligence for legal assistance. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 4–6.
- Bhattacharya, P., Paul, S., Ghosh, K., Ghosh, S., and Wyner, A. (2019b). Identification of rhetorical roles of sentences in indian legal judgments. In *Legal Knowledge and Information Systems: JURIX 2019: The Thirty-second Annual Conference*, volume 322. IOS Press.
- Bhattacharya, P., Poddar, S., Rudra, K., Ghosh, K., and Ghosh, S. (2021). Incorporating domain knowledge for extractive summarization of legal case documents. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 22–31.
- Cocarascu, O. and Toni, F. (2017). Identifying attack and support argumentative relations using deep learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1374–1379.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Eckle-Kohler, J., Kluge, R., and Gurevych, I. (2015). On the role of discourse markers for discriminating claims and premises in argumentative discourse. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2236–2242.
- Ghosh, K., Pawar, S., Palshikar, G., Bhattacharyya, P., and Varma, V. (2020). Retrieval of prior court cases using witness testimonies. In *Legal Knowledge and Information Systems*, pages 43–51. IOS Press.
- Jain, N. (2021). Customer sentiment analysis using weak supervision for customer-agent chat. *arXiv preprint arXiv:2111.14282*.
- Lawrence, J., Bex, F., Reed, C., and Snaith, M. (2012). Aifdb: Infrastructure for the argument web. In *Computational Models of Argument*, pages 515–516. IOS Press.
- Malik, V., Sanjay, R., Nigam, S. K., Ghosh, K., Guha, S. K., Bhattacharya, A., and Modi, A. (2021). ILDC for CJPE: Indian legal documents corpus for court judgment prediction and explanation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4046–4062. Online, August. Association for Computational Linguistics.
- Mallory, E. K., de Rochemonteix, M., Ratner, A.,

- Acharya, A., Re, C., Bright, R. A., and Altman, R. B. (2020). Extracting chemical reactions from text using snorkel. *BMC bioinformatics*, 21:1–15.
- Mandal, A., Ghosh, K., Ghosh, S., and Mandal, S. (2021). Unsupervised approaches for measuring textual similarity between legal court case reports. *Artificial Intelligence and Law*, pages 1–35.
- Menini, S., Cabrio, E., Tonelli, S., and Villata, S. (2018). Never retreat, never retract: Argumentation analysis for political speeches. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. 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*, pages 1003–1011.
- Palau, R. M. and Moens, M.-F. (2009). Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107.
- Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Poudyal, P., Savelka, J., Ieven, A., Moens, M. F., Gonçalves, T., and Quresma, P. (2020). ECHR: Legal Corpus for Argument Mining. In *Proceedings of the 7th Workshop on Argument Mining*, pages 67–75.
- Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., and Ré, C. (2017). Snorkel: Rapid training data creation with weak supervision. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 11, page 269. NIH Public Access.
- Reed, C., Mochales Palau, R., Rowe, G., and Moens, M.-F. (2008). Language resources for studying argument. In *Proceedings of the 6th conference on language resources and evaluation-LREC 2008*, pages 2613–2618. ELRA.
- REED, M. J. J. L. C. (2014). Ova+: An argument analysis interface. In *Computational Models of Argument: Proceedings of COMMA*, volume 266, page 463.
- Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.
- Stab, C. and Gurevych, I. (2017). Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.
- Stab, C., Daxenberger, J., Stahlhut, C., Miller, T., Schiller, B., Tauchmann, C., Eger, S., and Gurevych, I. (2018). Argumenttext: Searching for arguments in heterogeneous sources. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: demonstrations*, pages 21–25.
- Walker, M., Tree, J. E. F., Anand, P., Abbott, R., and King, J. (2012). A corpus for research on deliberation and debate. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 812–817.