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 Ltd., Pune, India.

Problem Definition

Input: A court judgment document $D = [s_1, s_2, \cdots, s_n]$ that contains a 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.

Proposed Approach

Linguistic rules to identify sentence pair having a Support or Attack relations.
(R1) Linguistic Rules based on Discourse markers,
(R2) Linguistic Rules based on Semantic Structure Overlap,
(R3) Distant Supervision using Argument Source

R1: Linguistic Rules based on Discourse Connectors

- Exploits multiple discourse connectors & the dependency tree structures.
- 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.`)
- Overlapping conditions also need to be satisfied between any two sentences.

Overlapping Conditions

- Overlap of at least K content words (NN, VB, or JJ (excluding stop words)) between the two sentences.
- Two words are overlapping: if they are exactly same or
  - if, have exactly same root form (e.g., `terminate` ⇔ `terminated`) or
  - if, are part of a single synset in WordNet (synonyms, e.g., `building` ⇔ `construction`) or
  - if, have high cosine similarity using GloVe word embeddings (e.g., `witness` ⇔ `testimony`).

R1: Examples

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} It seems that the Sessions Judge had directed production of carbon copy of some documents and written some letters (Exhibits C-1 to C-4)

s_i **However** , it was pointed out by the Public Prosecutor that the originals of Case Diary and the documents were already there before the Court and , therefore , there was no question of producing the carbon copy of the record

***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.

R2: Linguistic Rules based on Semantic Structure Overlap

- If two sentences are close and have a common semantic structure having at least one sentence Argumentative then there is a Support or Attack relations.
- Common Semantic Structure Overlap consists of a head (event noun or verb) with its corresponding arguments which are present for both the sentences.
- Both heads and arguments must satisfy the overlapping conditions.
- An argument is:
  - A descendant of the head in the dependency tree if the head is a verb, or
  - A descendant of the head's lowest verb ancestor if the head is a noun. (e.g., the lowest verb ancestor of the head conviction is stand)
- If the matching words (head or its arguments) are antonyms either by WordNet-based or satisfying negation rules then the common semantic structure is said to have opposite meaning in the two sentences.
- Negation pairs: e.g., `guilt`: `innocent` (Antonyms), `commit`: `not committed` (Explicit), `justice`: `failure of justice` (Implicit), `entertained`: `not only entertained` (Implicit)

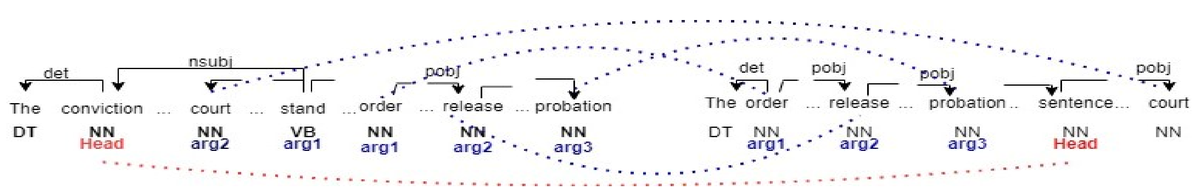

Figure: Example of Common Semantic Structure

R2: Examples

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`.
s_2: The `order` of `release` on `probation` is merely in substitution of the `sentence` to be imposed by the `Court`.
Head: `conviction` ⇔ `sentence` (identified as synonyms as per WordNet)
Arguments: `order`, `probation`, `court`, `release` (s_1 **Support** s_2)

s_1: The `learned` Single `Judge` of the `High` Court `committed` grave `error` by granting substantive relief to respondent No.1
s_2: `Learned` counsel for the respondents supported the impugned `judgment` and argued that the `High` `Court` did `not` `commit` any `error` by granting relief to respondent
Head: `committed` ⇔ `commit` (negated)
Arguments: `judgment`, `error`,`learn`, `high`, `court` (s_2 **Attack** s_1)

R3: Distant Supervision using Argument Source

- Identify sentences containing arguments by contesting parties (e.g., `Appellant` or `Respondent`) as well as lower courts mentioned in the court documents.
- Validate the above pairs by applying a common semantic structure routine.
- For Support Relation - Both sentences are stated by same party; For Attack Relation - Both sentences are stated by different party.

R3: Examples

s_1: The **appellants** contended that the High Court committed an error in relying on Ex.A2 as it was 3 years prior to the acquisition and there was a steep `increase` in the `value` of land during that period nearly 3 years prior to the acquisition.
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 - Appellant type and s_2 - Respondent type. The Common Semantic Structure is –
Head: `increase` ⇔ `increase` (exact match); **Argument**: `value`;(s_2 **Attack** s_1)

Results

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

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

- From the above proposed rules **20,506** positive sentence pairs (Support or attack relation) are constructed and other 20,000 negative sentence pairs (no Relation) are selected randomly. Further, using these constructed examples BERT model is trained.

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

Table: 5-fold cross-valid. results by the BERT-based sentence pair classification model

Conclusion

- We constructed the dataset of **20,506 sentence pairs** from court judgements that are labeled with Support or Attack relations.
- In future, we plan to improve the coverage and the precision of the existing rules.

Contact: ali.basit@tcs.com