Investigating Logic Tensor Networks for Neural-Symbolic Argument Mining

Andrea Galassi[1], Marco Lippi[2], Paolo Torroni[1]

[1]DISI, University of Bologna, Italy

[2]DISMI, University of Modena and Reggio Emilia, Italy

Problem Definition

Argument Mining

- Aim: automatic extraction of arguments and their relations from natural language texts
- S.o.t.a. approaches based on deep networks
- Still, debates and persuasion are still challenging domains for deep learning by itself

Neural-Symbolic Argument Mining

- Exploit domain and expert knowledge by imposing rules and constraints
- Problem: most of NeSy frameworks are under development and not optimized for AM tasks
  - **Collective classification**: must consider the network of connected components
  - Scalability: large datasets containing large documents
- **Objective**: classification of components (CLAIM or PREMISE) and link prediction exploiting a neural-symbolic framework to impose domain-related constraints

Logic Tensor Networks

Framework features

- Tensorization approach: FOL entities are embedded into real-valued tensors
- Background knowledge expressed through **first-order fuzzy logic**
- Vertical-hybrid system: high-level logic is placed on top of deep networks
- Possible to use any neural model (but there are limitation due to scalability issues)
- Possible to specify rules as FOL strings

LTN Entities

- **Variables**: abstract representation, linked to the possible groundings (real data)
- **Predicates**: operations over variables, produce a single value between 0 and 1
- **Axioms**: logic conditions that are used as optimization objectives

LTN for Argument Mining

Predicates

Define 4 predicates:

- PREMISE(X) and CLAIM(X)
- LINK(X,Y) and ~LINK(X,Y)
  - LINK(X,Y) means X supports Y

Networks

Define two networks:

- *NNComp* predicts components type (P or C)
- *NNLink* predicts relationships (yes or no)

Each output of each network is the probability of a class.

Combining Predicates and Networks

Association between degree of truth of a predicate and output of the networks

- Outputs of *NNComp* are connected to PREMISE and CLAIM
- Outputs of *NNLink* are connected to LINK and ~LINK

Method

Neural Models

Use of simple neural models

Compare two training methods

- *Neural* is trained only on data
- *NeSy* is trained combining data-driven and rules-driven optimization

Rules Axioms

Anti-symmetry:

If A=>B is true, then B=>A is false

$$\forall vC_1, vC_2 : LINK(vC_1, vC_2) \Rightarrow \sim LINK(vC_2, vC_1)$$

Claims can be linked only to claims:

If A=>B, and A is a claim, then B is a claim.

$$\forall vC_1, vC_2 : LINK(vC_1, vC_2) \wedge CLAIM(vC_1) \Rightarrow CLAIM(vC_2)$$

Results

Evaluation Criteria

- Classification accuracy
- Robustness against random
- Compliance to properties

Results

- The NeSy approach improves all the aspects, especially the compliance

Dataset	Approach	Classification Comp.	Classification Link	Agreement Comp.	Agreement Link	Properties Eq. 2	Properties Eq. 3
Neoplasm	NEURAL	79 - **80**	34 - 31	77	64	87 - 81	96 - 85
	NESY	79 - 78	**35 - 35**	**79**	**70**	**99** - 96	**99** - 94
Glaucoma	NEURAL	**82 - 82**	45 - 43	**75**	66	93 - 90	89 - 74
	NESY	81 - **82**	**47** - 45	**75**	**71**	**≈100** - 98	**99** - 90
Mixed	NEURAL	**81 - 81**	38 - 34	75	64	89 - 85	95 - 86
	NESY	**81** - 80	39 - **40**	**76**	**69**	**≈100** - 97	**97** - 96

AbstRCT dataset: scientific abstracts

Conclusion

Features

- Rules can be used at inference time to investigate models
- Using FOL rules: domain experts do not need machine learning expertise
- Decoupled architecture: symbolic and sub-symbolic parts are independent
- Improvement in performances

Open Challenges and Future Works

- Scalability! Currently impossible to use s.o.t.a. models or bigger datasets
- Rules with different granularity
- Weighted loss: rules vs preferences
- Use of predicates without grounding: Inference of new properties

StarAI@IJCLR 2021 - Tenth International Workshop on Statistical Relational AI