1 Introduction

1.1 Explainability in NLP

1.1.1 What Is Explainability

1.1.2 Why Is Explainability Important

1.1.3 Properties of Explanations

1. Time
2. Model accessibility
3. Scope
4. Unit of explanation
5. Form of explanation
6. Target audience

1.1.4 Principles of Explanations

1. Faithfulness
2. Plausibility
3. Input Sensitivity
4. Model Sensitivity
5. Completeness
6. Minimality

1.2 Faithfulness as a Principle

1.2.1 Definition

1.2.2 Relation between Faithfulness and Other Principles

1. Faithfulness vs. Plausibility
2. Faithfulness vs. Sensitivity, Implementation Invariance, Input Invariance, and Completeness
3. Faithfulness vs. Minimality

1.2.3 Importance

1.2.4 Evaluation

1. Axiomatic evaluation
2. Predictive power evaluation
3. Robustness evaluation
4. Perturbation-based evaluation
5. White-box evaluation
6. Human perception evaluation

2 Attempts at Faithful Explanation

2.1 Overview with Motivating Example

2.2 Similarity Methods

1. (Caruana et al., 1999)
2. (Wallace et al., 2018)
3. (Rajagopal et al., 2021)

2.3 Analysis of Model-Internal Structures

1. The pre-attention era
    (a) (Karpathy et al., 2015)
    (b) (Li et al., 2016)
    (c) (Strobelt et al., 2018)
    (d) (Poerner et al., 2018)
    (e) (Hiebert et al., 2018)
    (f) Tools: RNNvis (Ming et al., 2017), LSTMVis (Strobelt et al., 2018), Seq2Seq-Vis (Strobelt et al., 2019)
2. The post-attention era
    (a) Attention as an explanation
        i. (Vig, 2019)
        ii. (Martins & Astudillo, 2016)
        iii. (Xie et al., 2017)
        iv. (Mullenbach et al., 2018)
        v. (Clark et al., 2019)

(b) Debate
   i. (Jain & Wallace, 2019)
   ii. (Wiegreffe & Pinter, 2019)
   iii. (Pruthi et al., 2020)
   iv. (Voita et al., 2019)
   v. (Raganato & Tiedemann, 2018)
   vi. (Voita et al., 2019)
   vii. (Ferrando & Costa-jussà, 2021)
   viii. (Bastings & Filippova, 2020)

(c) How to make attention more faithful
   i. (Tutek & Snajder, 2020)
   ii. (Hao et al., 2021)

(d) Tools: BertViz (Vig, 2019), LIT (Tenney et al., 2020)

2.4 Backpropagation-based Methods

1. Gradient methods
   (a) Simple Gradients (Baehrens et al., 2010; Simonyan et al., 2014)
   (b) Gradient×Input (Denil et al., 2015)
   (c) Integrated Gradients (Sundararajan et al., 2017)
   (d) SmoothGrad (Smilkov et al., 2017)
2. Propagation methods
   (a) DeconvNet (Zeiler & Fergus, 2014)
   (b) Guided BackPropagation (Springenberg et al., 2015)
   (c) Layerwise Relevance Propagation (Bach et al., 2015)
   (d) DeepLift (Shrikumar et al., 2017)
   (e) Deep-Taylor Decomposition (Montavon et al., 2017)
3. Tools: AllenNLP Interpret (Wallace et al., 2019), Captum (Kokhlikyan et al., 2020), RNNbow (Cashman et al., 2018), DeepExplain (https://github.com/marcoancona/DeepExplain)

2.5 Counterfactual Intervention

1. Intervening in inputs
   (a) Feature-targeted intervention
      i. Feature-targeted erasure
         A. Leave-one-out (Kádár et al., 2017; Li et al., 2017)

B. Subsets of features: Anchors (Ribeiro et al., 2018), DiffMask (De Cao et al., 2020)
C. Surrogate models: LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017)
D. Feature interactions: Archipelago (Tsang et al., 2020)

ii. Feature-targeted perturbation
A. Counterfactual examples: (Kaushik et al., 2020; Wu et al., 2021)

(b) Example-targeted intervention
i. Influence functions (Han et al., 2020; Koh & Liang, 2017)

2. Intervening in model representations
(a) Neuron-targeted intervention
i. Neuron-targeted erasure
A. Leave-one-out (Bau et al., 2019; Li et al., 2017)
ii. Neuron-targeted perturbation
A. Causal mediation analysis (Vig et al., 2020)
(b) Feature-representation-targeted intervention
i. Feature-representation-targeted erasure
A. Amnesic Probing (Elazar et al., 2021)
B. CausalLM (Feder et al., 2021)
ii. Feature-representation-targeted perturbation
A. AlteRep (Ravfogel et al., 2021)
B. (Tucker et al., 2021)

3. Tools: Captum (https://captum.ai), LIT Tenney et al., 2020, LIME Ribeiro et al., 2016, SHAP Lundberg and Lee, 2017, Anchors Ribeiro et al., 2018, Seq2Seq-Vis Strobelt et al., 2019, the What-if Tool Wexler et al., 2020

2.6 Self-Explanatory Models

1. Explainable architecture
(a) Neural Module Networks
i. (Andreas et al., 2016b)
ii. Dynamic Neural Module Network (Andreas et al., 2016a)
iii. End-to-End Module Network (Hu et al., 2017)
iv. (Y. Jiang et al., 2019)
v. (Gupta et al., 2019)
(b) Neural-Symbolic Models
i. Neural-Symbolic VQA (Yi et al., 2018)

            ii. Neuro-Symbolic Concept Learner (Mao & Gan, 2019)
   (c) Models with constraints
      i. (Alvarez Melis & Jaakkola, 2018)
      ii. (Schwartz et al., 2018)
      iii. (Deutsch et al., 2019)
      iv. (C. Jiang et al., 2020)

2. Generating explanations
   (a) Predict-then-explain
      i. (Hendricks et al., 2016)
      ii. (Camburu et al., 2018)
      iii. (Park et al., 2018)
      iv. (Kim et al., 2018)
   (b) Explain-then-predict
      i. An extract from the input
         A. (Lei et al., 2016)
         B. (Bastings et al., 2019)
         C. (Jain et al., 2020)
         D. (Jacovi & Goldberg, 2021)
      ii. Natural language
         A. (Camburu et al., 2018)
         B. (Camburu et al., 2020)
         C. NILE variant (Kumar & Talukdar, 2020)
   (c) Jointly-predict-and-explain
      i. (Ling et al., 2017)
      ii. wT5 (Narang et al., 2020)
      iii. ProofWriter (Tafjord et al., 2021)
      iv. EntailmentWriter (Dalvi et al., 2021)
      v. (Rajani et al., 2019)
      vi. NILE variant (Kumar & Talukdar, 2020)

3 Summary and Discussion

3.1 Virtues

3.2 Challenges and Future Work

4 Conclusion