TriplEx: Triple Extraction for Explanation

Luca Moroni

Università di Pisa - Data Mining

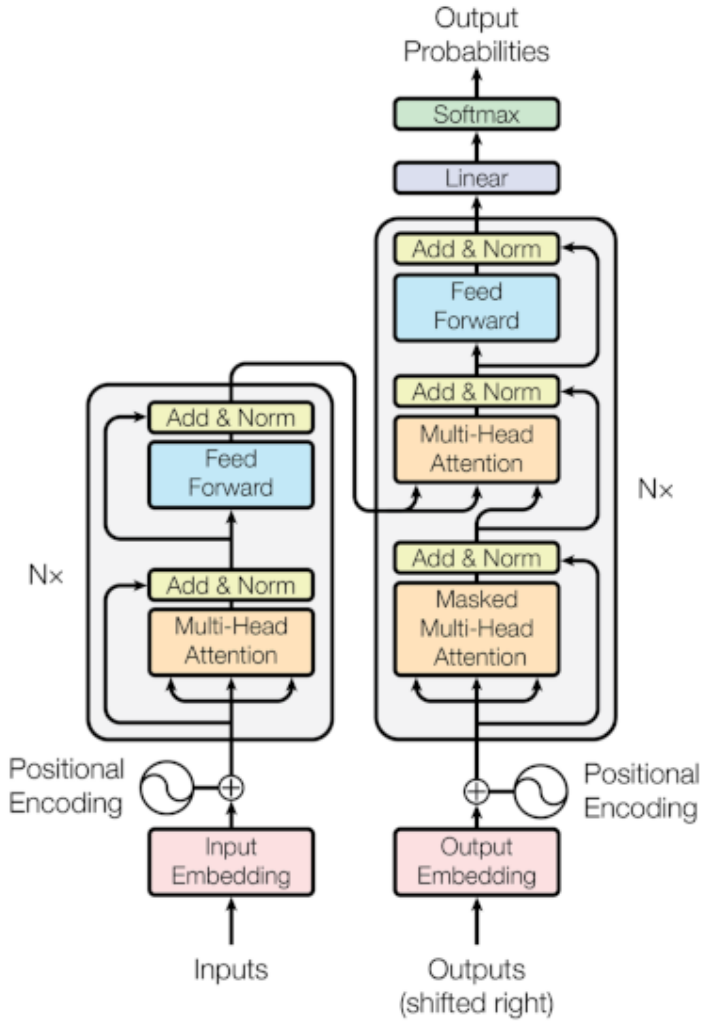
2022

Background: Transformers

Attention formulae.

$$Att_h(K, Q, V) = softmax(\frac{QK^T}{d})V_h,$$

where d is a normalization parameter, and V_h is a set of learned parameters mapping the keys and values to differ reppresentations.



Motivation and Context

Transformer-based models have reached a lot of success nowadays becoming a de-facto standard, tending to be large models with millions if not billions of parameters, making them black boxes to anyone who tries to understand their predictions.

Explainable AI (XAI) offers two kind of algorithmic solutions,

- Token Importance (TI),
- Natural Language Explanation (NL).

TI methods need the input permanence assumption even in the phase of explanation.

NL methods rely on a generator model, sometimes lacking in semantic consistence.

Motivation and Context

A goal is to minimize the reliance on external language model (NL) during the explanation and assumption on the input persistence (TI).

The work done in [1] aims to generate self-contained explanations, while trying to emulate neural reasoning abstraction and similarity, with the usage of structured Knowledge Base, for Transformer-based models trained on NLI, STS or TC tasks.

- **Natural Language Inference**: $\{(p_i, h_i), y_i\}$
- **Semantic Text Similarity**: $\{(s_{i,1}, s_{i,2}), y_i\}$
- **Text Classification**: $\{x_i, y_i\}$

Method: Algorithm

TriplEx locally explain predictions in the form of triples.

TriplEx consist in a three-stage pipeline,

- **Information Extraction**,
- **Candidates Generation and Selection**,
- **Explanation Enrichment, Alignment score**.

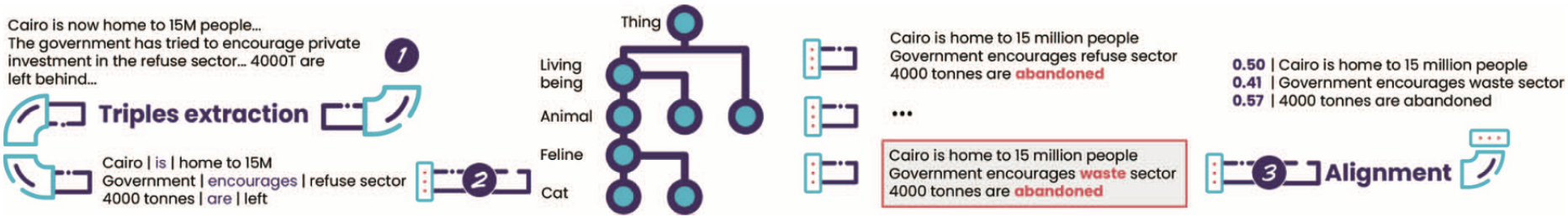


Figure: An example of the TRIPLEX pipeline on a NLI task

Method: Information Extraction

As first Step of Information Extraction the algorithm extract a set of triple-like propositions e from the input.

This step yields a baseline explanation stub e.

For NLI the input is the premise. On the other hand for STS and TC the input is the whole text.

Information extraction methods,

- **OpenIE**,
- **OIIIE**,
- **ClausIE**.

Method: Candidates Generation and Selection

As second step, the algorithm perturb the stub e through the use of a knowledge base , either by generalization using (NLI task) or similarity (STS, TC tasks), leveraging on **hypernyms** and **synonyms** taxonomy. **WordNet** is the used KB.

This process generate a set of candidate explanations from the stub, then the candidates are selected out.

Given the hypernyms distance d_h as the number of level between two concepts in the KB, in TriplEx is used the d_h to select the most abstract candidate explanation. (the same for synonyms).

Method: Explanation Enrichment, Alignment score

Denoting by f the transformer based model, in an input text x_i we can indicate with $\alpha_i(a, b)$ the attention weight of f between the two tokens, we can extend this information to set of tokens T_e, T_x.

$$\alpha_i(T_e, T_x) = \frac{\sum_{a \in T_e, b \in T_x} \alpha_i(a, b)}{|T_e| + |T_x|}.$$

Additionally the attention is addressed in a given head h of a specific layer l.

T_e are triples from the explanation, while T_x are tokens from the input text.

Algorithm I

Input: Input x, maximum depth K, search radius γ, knowledge base G, strategy S
Output: Expanded terms T

1: **function** EXPAND(x, K, γ, G, S)
2: **if** $S ==$ 'hypernyms' **then**
3: $x' \leftarrow x[0]$ ▷ select only the premise
4: $T \leftarrow \text{HYPERNYMS}(x', K, \gamma, G)$ ▷ create hypernyms
5: **else**
6: $T \leftarrow \text{SYSNONYMS}(x, K, \gamma, G)$ ▷ create synonyms
7: **return** T

Figure: Algorithm 1, Expansion algorithm.

Algorithm II

Input: Input $x \in \{(p_i, h_i), t_i\}$, Information extractor I, Transformer-based model f, knowledge graph G, maximum depth K, search radius γ, head h, layer l, Strategy S
Output: explanation e, alignment score a

function TRIPLEX(t, f, K, γ, G, S)
  $y \leftarrow f(x)$
  $\widetilde{E} \leftarrow \emptyset$
  $e \leftarrow \mathrm{I}(t, S)$ ▷ create explanation stub
  $H \leftarrow \mathrm{EXPAND}(e, K, \gamma, G, S)$ ▷ create explanation stub according to the task
  $\widetilde{E} \leftarrow \mathrm{PERTURB}(e)$ ▷ perturb
  candidates $\leftarrow []$
  distances $\leftarrow []$
  for $\widetilde{e} \in \widetilde{E}$ **do**
    $\widetilde{y}_i \leftarrow f(\widetilde{e})$
    $d_{\widetilde{e}} \leftarrow \mathrm{distance}(G, e, \widetilde{e})$ ▷ compute candidate distance
    if $\widetilde{y}_i = y$ **then** ▷ select explanations
      candidates $\leftarrow \mathrm{APPEND}(\text{candidates}, \widetilde{e})$ ▷ add explanation
      distances $\leftarrow \mathrm{APPEND}(\text{distances}, d_{\widetilde{e}})$ ▷ add explanation distance
  $e^* \leftarrow \text{candidates}[\arg\max_t \text{distances}[t]]$ ▷ select explanation
  if $S ==$ 'hypernyms' **then**
    for $e_t \in e^*$ **do**
      $a_t \leftarrow \alpha_i^{h,l}(e_t, x_i[1])$ ▷ alignment to hypothesis
  else
    for $e_t \in e^*$ **do**
      $a_t \leftarrow \alpha_i^{h,l}(e_t, x)$ ▷ alignment to whole input
  return e^*, a

Figure: Algorithm 2, Explanation algorithm

Algorithm

Premise

Cairo is now home to some 15 million people – a burgeoning population that produces approximately 10,000 tonnes of rubbish per day, putting an enormous strain on public services. In the past 10 years, the government has tried hard to encourage private investment in the refuse sector, but some estimate 4,000 tonnes of waste is left behind every day, festering in the heat as it waits for someone to clear it up. It is often the people in the poorest neighborhoods that are worst affected. But in some areas they are fighting back. In Shubra, one of the northern districts of the city, the residents have taken to the streets armed with dustpans and brushes to clean up public areas which have been used as public dumps.

Hypothesis

15 million tonnes of rubbish are produced daily in Cairo.

Explanation

Alignment	Rank	Subject	Predicate	Object
.050	2	Cairo	is	home to some 15 million people
.041	5	Government	encourage	finance in waste sector
.043	4	Finance	is	in waste sector
.046	3	People	are	in poor neighborhood
.057	1	4000 tonnes	are	left

Figure: Example in a NLI scenario

Experiments

The experiments done in [1] want to understand if the proposed method is concise and coherent.

TriplEx was evaluated on four NLI datasets (GLUE, SuperGLUE, RTE and MNLI), for STS and TC datasets Amazon Polarity and Semantic Text Similarity are used.

	#Records	Performance
RTE	276	93.0
MNLI	9814	91.4
AX_g	355	92.7
AX_b	1103	53.2
AMAZON	2530	92.8
STS	1004	92.9

Figure: Performances of the fine-tuned models

Experiments

To Evaluate Complexity they estimate through the ratio between the explanation and premise/input text length.

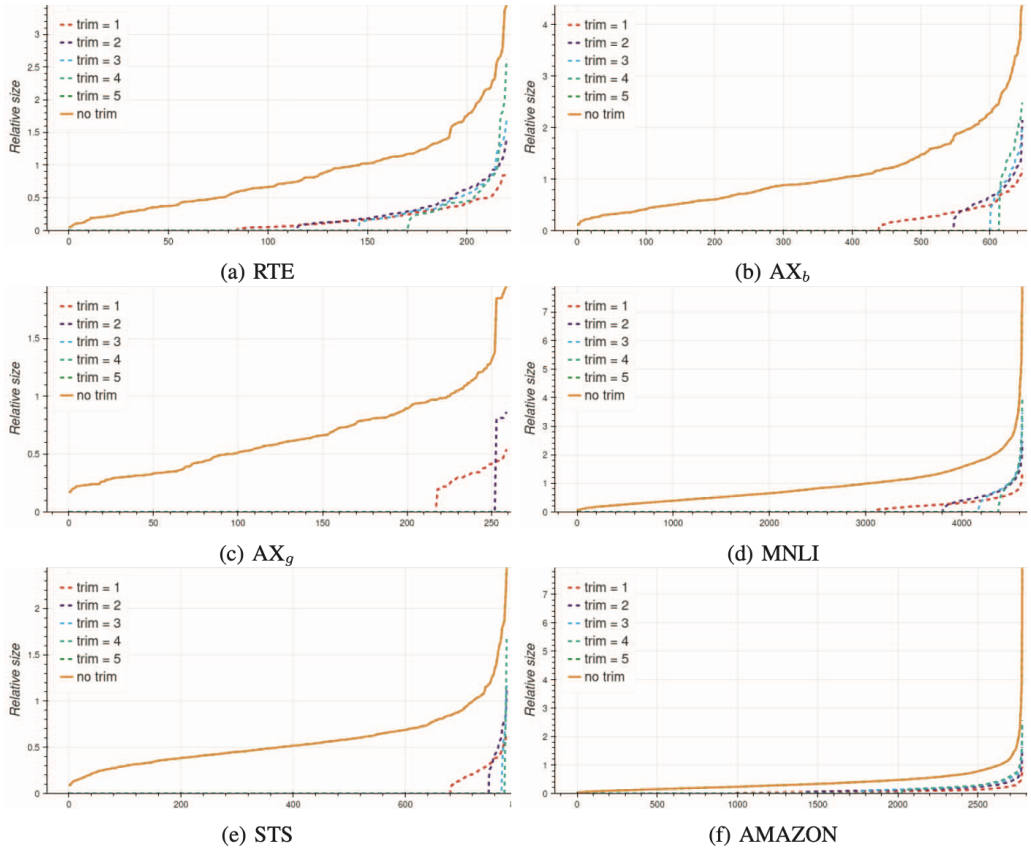


Figure: Explanation length for complete and trimmed (dashed line) explanation

Experiments

To evaluate Similarity they report cosine similarity between TriplEx and a baseline, and between TriplEx and the whole input text.

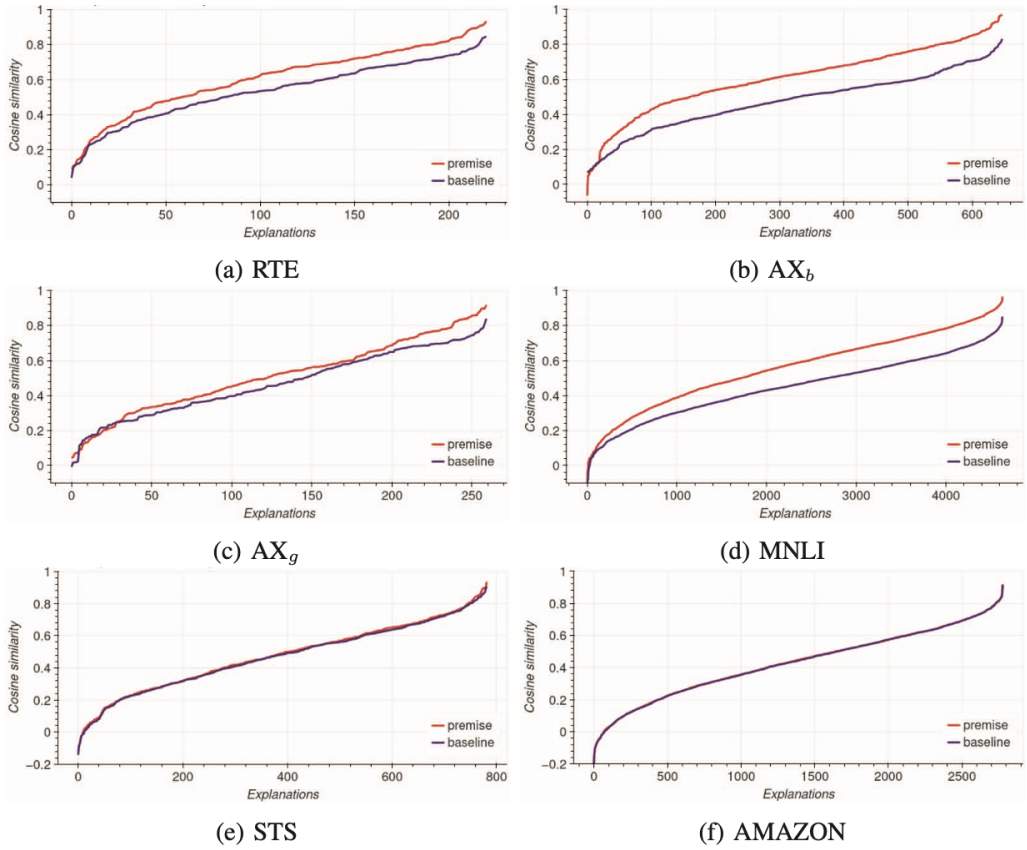


Figure: Cosine similarity between i) TriplEx and a Baseline ii) Triplex and sentence embedding

Experiments

To evaluate Perplexity they report sorted perplexity measures on input text and explanation, as computed by GPT-2.

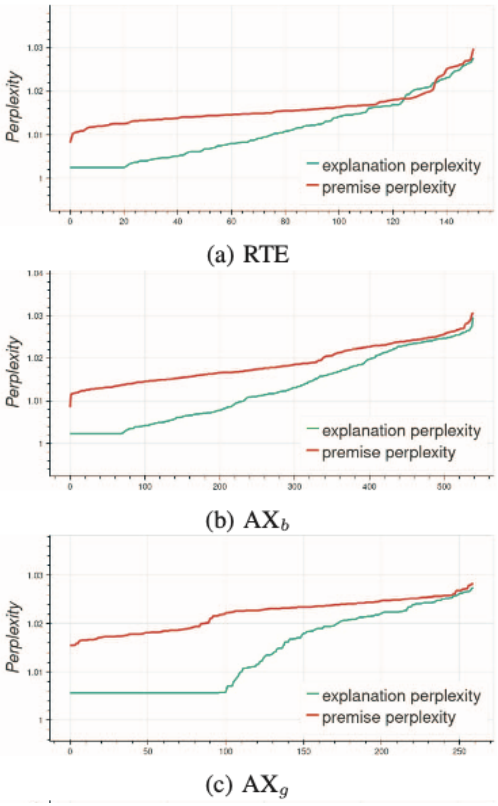


Figure: Explanation and premise/text perplexity i)

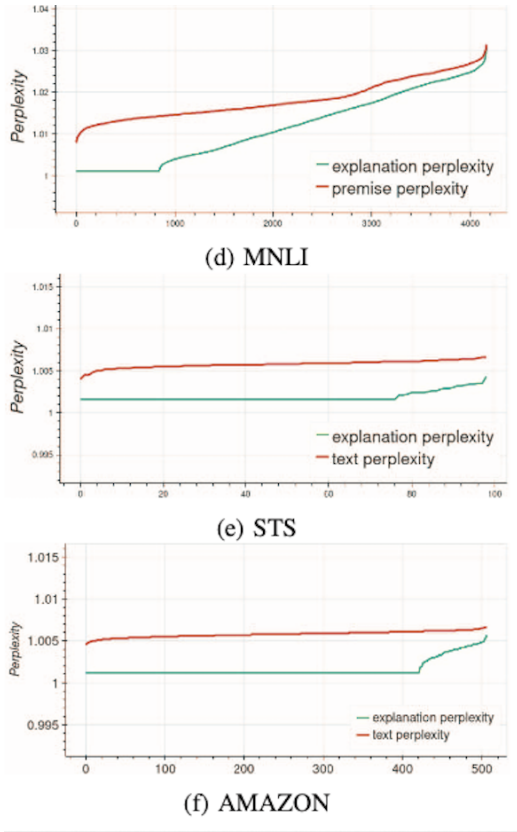


Figure: Explanation and premise/text perplexity ii)

Conclusions

Unlike other approaches TriplEx removes dependence from external text generator models.

Moreover in Triplex, explanations show relatively low complexity and are high similar to baseline existing approaches, yet they met some limit cases in which explanation complexity degenerates (see **figure pg. 13**).

As stated in [1], they aim to address such limit cases and to further reduce explanation complexity by properly masking triples that do not contribute to the prediction of the model.

Bibliography

[1] Mattia Setzu, Anna Monreale, and Pasquale Minervini. "TRIPLEx: Triple Extraction for Explanation". In: *2021 IEEE Third International Conference on Cognitive Machine Intelligence (CogMI)*. IEEE. 2021, pp. 44–53.