Boosting Logical Fallacy Reasoning in LLMs via Logical Structure Tbee

Yuanyuan Lei and Ruihong Huang Department of Computer Science and Engineering Texas A M University, College Station, TX yuanyuan, huangrhGtamu.edu

February 21, 2026

Abstract

educational essay (Jin et a1.,2022), political de-
Logical fallacy uses invalid or faulty reasoning in the construction of a statement. Despite the prevalence and harmfulness of logical fallacies, detecting and classifying logical fallacies still remains a challenging task. We observe that logical fallacies often use connective words to indicate an intended logical relation between two arguments, while the argument semantics does not actually support the logical relation. Inspired by this observation, we propose to build a logical structure tree to explicitly repre-
sent and track the hierarchical logic flow among relation connectives and their arguments in a statement. Specifically, this logical structure tree is constructed in an unsupervised manner guided by the constituency tree and a taxonomy of connectives for ten common logical relations, with relation connectives as non-terminal nodes and textual arguments as terminal nodes, and the latter are mostly elementary discourse units.
We further develop two strategies to incorporate the logical structure tree into LLMs for fallacy reasoning. Firstly, we transform the tree into natural language descriptions and feed the tex- tualized tree into LLMs as a part of the hard text prompt. Secondly, we derive a relation-aware tree embedding and insert the tree embedding into LLMs as a soft prompt. Experiments on benchmark datasets demonstrate that our ap- proach based on logical structwe tree signifi- cantly improves precision and recall for both fallacy detection and fallacy classiflcation 1.

1 Introduction

Logical fallacy refers to the use of invalid or flawed reasoning in an argumentation (Risen et a1.,2007; Walton, 2010; Cotton, 2018). Logical fallacy can occur as unintentional mistakes or deliberate per- suasions in a variety of human communications, such as news media (Da San Martino et a7.,2019),

lThe code and data link is: https://github.com/yuanyua n le i - nIp/Iogi caI-fa I lacy-emnl p-2024

Logical fallacy uses invalid or faulty reasoning in the construction of a statement. Despite the prevalence and harmfulness of logical fallacies, detecting and classifying logical fallacies still remains a challenging task. We observe that logical fallacies often use connective words to indicate an intended logical relation between two arguments, while the argument semantics does not actually support the logical relation. Inspired by this observation, we propose to build a logical structure tree to explicitly represent and track the hierarchical logic flow among relation connectives and their arguments in a statement. Specifically, this logical structure tree is constructed in an unsupervised manner guided by the constituency tree and a taxonomy of connectives for ten common logical relations, with relation connectives as non-terminal nodes and textual arguments as terminal nodes, and the latter are mostly elementary discourse units. We further develop two strategies to incorporate the logical structure tree into LLMs for fallacy

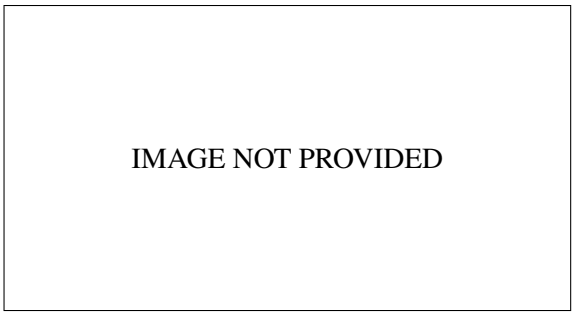

Figure 1: Figure 1: Examples of logical fallacy sentences and their logical strucrure trees. The logical structure tree features logical relation connectives as non-terminal nodes, and textual arguments as terminal nodes.

reasoning. Firstly, we transform the tree into natural language descriptions and feed the tex- tualized tree into LLMs as a part of the hard text prompt. Secondly, we derive a relation-aware tree embedding and insert the tree embedding into LLMs as a soft prompt. Experiments on benchmark datasets demonstrate that our ap- proach based on logical structwe tree significantly improves precision and recall for both fallacy detection and fallacy classiflcation 1.

The key observation is that logical fallacies heavily rely on connective phrases to indicate an in- tended logical relation between two textual argu- ments, while the semantics of the arguments do not actually support the logical relation. Figure 1 shows two examples where the connec- tive phrases were bolded. The first example uses the connective words therefore and cause to sug- gest a causal relation between vaccinations and increasing flu cases, however, the temporal relation between the two events as stated in the flrst half of the statement does not necessarily entail a causal relation between them, and indeed, their seman- tics do not actually support the suggested causal relation. Recognizing this discrepancy undermines the credibility of the whole statement. Similarly in the second example, the connective word likewise is commonly used to indicate an analogy relation, however, the second argument is clearly a specific case of the general condition stated in the first ar- gument and therefore there is no analogy relation between them, and recognizing this mismatch be- tween the suggested logical relation and the real relation enables us to detect this fallacy.

Therefore, we propose to construct a logical structure tree that organizes all connective phrases in a statement and their texfual arguments into a hierarchical structure. We expect the logical struc- ture fee to effectively capture the juxtaposition of connective phrase suggested logical relations and the real logical relations between textual ar- guments, and therefore guide LLMs in fallacy de- tection and classification. Specifically, a logical structure fee consists of relation connectives as non-terminal nodes and textual arguments as ter- minal nodes, and the latter mostly coffesponds to elementary discourse units (EDU) considered in discourse parsing. Figure I shows the logical struc- ture trees constructed for the two example texts.

As the logical relation indicated by a connective phrase may not be supported by semantics of its arguments in the context, we identify the purpose- fully indicated logical relations in a context-free unsupervised manner by matching a connective phrase with a taxonomy of connectives compiled for ten corlmon logical relations (conjunction, al- ternative, restatement, instantiation, contrast, con- cession, analogy, temporal, condition, causal). To construct a logical structure tree, we first construct a constituency tree for a statement and then search in the constituency tree for connective phrases in the top-down left to right order, and the first found connective phrase will be the root node of the logi- cal structure tree. Next, we identify the text spans of its two arguments using rules and recursively build the left and right sub-trees by applying rhe same procedure to constituency tree segments cor- responding to the two arguments.

The logical structure tree is integrated into LLMs for fallacy reasoning using two strategies. The first considers textualized tree, where we convert the tree into natural language descriptions, making the tree readable by LLMs. Particularly, we describe the relations and arguments in a bottom-up man- ner, providing the LLMs with insight into logical relations from a local to global perspective. We then concatenate the textualized tree with the in- struction prompt, and input them into LLMs as a hard prompt. The second considers tree-based soft prompt, where we derive a relation-aware tree em- bedding. Specifically, we design relation-specific encoders to process each type ofrelation and incre- mentally derive the tree embedding from bottom up to the root node. We then insert

the tree embed- ding into LLMs as a soft prompt for further tuning. Experiments on benchmark datasets across vari- ous domains and genres validate that our approach based on logical structure tree effectively improve precision and recall for both fallacy detection and fallacy classification tasks. Our main contributions are summarized as follows:

. We propose to construct a logical structure fee to captwe the juxtaposition of connective phrase suggested logical relations and the real logical relations between textual arguments, and use it to serve as additional guidance for fallacy detection and classification.

. We effectively improve the Fl score for fallacy detection by up to 3.457o and fallacy classifi- cation by np to 6.7SVo across various datasets.

2 Related Work

Logical Fallacy is erroneous patterns ofreasoning (Walton, 1987; Fantino er al., 2003). Initial work explored the taxonomy of fallacies (Tindale,2007; Greenwell eta1.,2006; Walton et al., 2008). Recent works have focused on the automatic detection and classification of fallacies. Habernal et al. (2017) developed a software that deals with fallacies in question-answering. Sheng et al. (2021) investi- gated ad hominem fallacy in dialogue responses. Habernal et al. (2018) explored the ad hominem fal- lacy from web argumentations. Stab and Gurevych (2017) recognized insufficient arguments in argu- mentation essays. Goffredo et al. (2022) catego- rized fallacies in political debates. Nakpih and Santini (2020) focused on fallacies in legal argu- mentations. Musi et al. (2022) researched fallacies about pandemics on social medias. (Alhindi et a1., 2022) proposed a multi-task prompting approach to learn the fallacies from multiple datasets jointly. Jin et al. (2022) proposed a structure-aware method to classify fallacies. Different from Jin et al. (2022) that masked out content words to form a sequence- based pattern, our paper proposes a tree-based hi- erarchical logical structure to unify both relation connectives and content arguments together.

Logical Reasoning abilities of large language models are gaining increasing research attention (Xu eta1.,2023; Chen et a1.,2021; Creswell eta1.,2022; Pi et al., 2022; Jrao et a1.,2022; Zhot et a1.,2023; Sanyal et a1.,2023; Parmar et a1.,2024). Olausson et al. (2023) combined large language models with flrst-order logic. Pan et al. (2023); Zhang et a7. (2023) empowered large language models with symbolic solvers. Pi et al. (2022) presented an ad- versarial pre-training framework to improve logical reasoning. Zhao et al. (2023) incorporated multi- step explicit planning into the inference procedure. Jiao et al. (2022) proposed a contrastive learning approach to improve logical question-answering. Different from these previous work, we particularly focus on logical fallacy reasoning, aiming to detect and classify fallacies.

Misinformation refers to the unverified or false information (Guess and Lyons, 2020; Armitage and Vaccari, 2021; Aimeur et al., 2023;Lei et al., 2024b). Misinformation detection was studied for years, such as fake news (Rashkin et al., 2017;Lei and Huang, 2023b; Oshikawa et a1.,2020), mmor (Ma et al., 2018; Li et al., 2019), satire (Yang et al., 2017), political bias (Lei et a1.,2022; Feng et a1., Z23;Devatine et a1., 2023;Lei and Huang, 2024), propaganda (Da San Martino et a1.,2019,2020; Lei and Huang, 2023a). Logical fallacies are often employed within misinformation to present invalid claim as credible, facilitating the spread of misin- formation (Beisecker eta1.,2024; Pauli et a—,2022; Bonial et a1.,2022). Developing automatic mod- els to detect logical fallacies can also benefit the identiflcation and mitigation of misinformation.

3 Logical Structure T[ee

The logical structure tree consists of relation con- nectives as non-terminal nodes, and textual argu- ments as terminal nodes. The relation connectives serve as parent nodes, and the two corresponding ar- guments are linked as left and right children nodes. Figure 1 illustrates examples of the logical structure tree. The logical structure tree is constructed in an unsupervised manner, guided by the constituency tree and a taxonomy of connectives complied for ten common logical relations.

TABLE CONTENT [ILLEGIBLE]

Table 1: Table l: The ten types of logical relations and their relation connectives.

3.1 RelationConnectives

The logical fallacies usually rely on relation con- nectives to indicate a logical relation. Inspired by the discourse relations proposed by Prasad et al. (2008), we deflne a taxonomy of ten logical rela- tions which are commonly seen: conjunctton, al- t e rnativ e, re st atement, i n s t an ti ati on, c ontra s t, c on- cession, analogy, temporal, condition, and causal relations. Moreover, we build a set of connective words and phrases that correspond to each type of logical relation, as shown in Table 1. This set of connectives includes the explicit discourse con- nectives from the PDTB discourse relation dataset (Prasad et al., 2008), and is further expanded by manually adding relevant connectives from the de- velopment set of the logic fallacy dataset (Jin et al., 2022).

We further conduct a statistical analysis on the distribution of ten logical relations and compare distributions betw een fall acy and n o fall a cy clas ses as well as across different fallacy classes, with the detailed results shown in Appendix A. The statis- tical analysis shows that both the fallacy and no fallacy classes contain many connective phrases and their distributions of the ten logical relations are also very similar. But as expected, different fal- lacy types tend to employ varying logical patterns, for example, False Dilemma uses more altemative relation, while Deductive Fallacy uses more anal- ogy relation.

3.2 Tiee Construction Algorithm

To construct a logical structure tree T1on4., we first construct a constituency ttee T.on for a statement. We use the stanza library2 to get the constituency

2https, //stanfordnlp. github. io/ stanza / const ituency . html

tree (Qi et a1., 2020). At the beginning, T1,na" is ini- tialized as an empty tree. Then we traverse the con- stituency treeT"on from top to bottom and from left to right, and match relation connectives within each subtree of Toon. If there is a subtree Scon(-) whose text equals to a relation connective u, we use the algorithm in section 3.3 to extract the two textual arguments a, B associated with tr.r. Then a new log- ical subtree Sbsu(u) is created, with the matched relation connective ?/ as a parent node, and the two arguments a) p as its left and right children. This new logical subtree Sbsi,c(w) is added into the log- ical structure tree T1oni". If the textual arguments a, B still contain other relation connectives, then we recursively match relation connectives in the arguments and replace the original argument node intheT1ona. with the newly created logical subtree. The termination condition is that all the relation connectives in the given text have been matched.

3.3 lbxtual Arguments Extraction

The textual arguments are the two content com- ponents linked by a relation connective. Given a matched relation connective tr.r, its corresponding subtree inlheT"on is ,9-r1-y. To extract the argu- ments of ?r, we find the parent tree of S *n61 in the T*n, denoted as P(,9"rr1ry). The text enclosed by P(S*"¡.)) is the concatenation of all its leaf node texts. If the text enclosed by parent tree P (S "on1*1) contains content before and after the relation con- nective u), i,e., has the form of a I w * d, then the left argument of tl is o and the right argument is B. If the text enclosed by parent tree P(S"on61) only contains content after the relation connective 't!, i.e., has the form of w + P, then the right ar- gument of tu is 0, arrd the left argument o is the text enclosed by grandparent tree P(P(5""""6)) subtracted by the text enclosed by P(5*"6).

4 Logical Fallacy Reasoning

We further design a framework to incorporate the logical structure tree into LLMs for fallacy detection and classification. This framework consists of two main components. The first is textualized tree, where we convert the logical structure tree into natural language descriptions, and feed it into LLMs as a hard text prompt. The second is tree-based soft prompt, where we derive a relation-aware tree embedding, and insert it into LLMs as a soft prompt for additional tuning. The hard and soft prompts are complementary: the hard prompt enriches the instruction with logical structure information, while

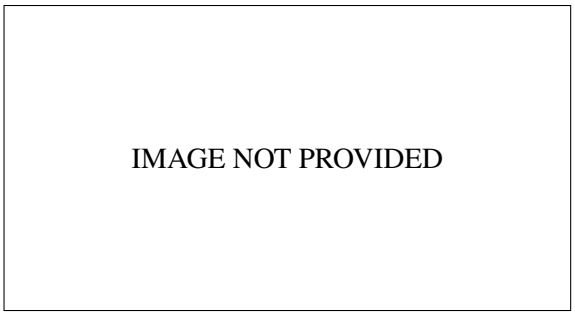

Figure 2: Figure 2: An illustration of logical fallacy classiflcation informed by logical structure tree.

the soft prompt facilitates direct tuning on tree embeddings. Figure 2 shows an illustration.

4.1 Textualized Tbee

The textualizedtree aims to transform the logical structure tree into the textual form, which can be interpretable by LLMs. As shown by the upper path of Figure2,the textualized tree is represented as a table which consists of three columns: left ar- gument, relation connective, right argument. Each row in the table represents a triplet (left argument, re lation c onn e c tiv e, ri g ht ar g umenr) correspondin g to each logical relation in the tree. In particular, we organize the triplets into the table in a bottom-up order, to provide the LLMs with insight into logical relations from a micro to macro perspective. The textualized tree is then input into the LLMs as a

Ilstuctio! prmpt Plc6r clusify the fillacy typ€ of the T*t. Choose one mwer ftom these frllocy g1rcs: <fallacy types li8>. Thc dcfnitiom of erch filley B?€ re I follom: <fallary tlpes defEiti@>Tcxti <fallacy ta(>.

part of the hard text promt: ht : T ert Embedd,er (tertuali, ze(T1oni.)) (1) where tertualize(.) denotes the textualization op- eration, TentUmbedder reders to the text embed- ding layer of LLMs, ht is the mapped embedding of the textualized tree.

4.2 Tfee-based Soft Prompt

The tree-based soft prompt is a tree embedding which is projected into LLMs as a soft prompt for further tuning. As shown by the lower path pf Figure 2, this process includes a tree encoder to derive the tree embedding, as well as a projection layer to transform the free embedding into the s;; representation space of LLMs.

During the tee encoder stage, we aim to derive a relation-aware tree embedding. To integrate re- lation information into free embedding, we design relation-specific encoders to process each type"of logical relation. For a simple tree whose children nodes are leaf nodes without hierarchical layers, its embedding is computed as:

e, : W'(et 0 e" @ e,) + b' (2)

where e, is the embedding of this simple tree, e6 ec, er are the embeddings of left argument, relation connective, and right argument, which are initialized as the average of word embeddings de- rived from RoBERTa language model (Liu et a1., 2019I@ denotesfeatureconcatenation,Wr,V are the trainable parameters of the encoder that corre- sponds to the relation type r, where 147r E p1d'xd', W" b, e Rd, and d : 768 is the dimension of em- bedding space in RoBERIa. The relation type r is one of the ten logical relations associated wittr the relation connective.

For the tree with hierarchical structure, we derive the tree embedding incrementally, starting from the bottom simple free and up towards the root node:

et : W'(d @ e" @ €,) + b' (3)

where el is the hee embedding, d; is the embedding of the left subfree, d, is the embedding of the right subfree, e" is the connective embedding.

During the projection stage, we transform the tree embedding e, into the same representation space of LLMs ttrough a projection layer, which includes two layers of neural networks:

6t: Wz(wr"r+ br) + a, g)

wfgre Wr, Wz, b1, b2 arc the nainable parameters s;; 9f ttre nrorygtion layer, W1 e Rdxd ,Wz e 4d! xd , b1,b2 e Rd" d is dimension of hidden states in RoBERTa, d/ is the dimension of embedding space of the target LLM. d1 is the resulting tree-based soft prompt, which is then inserted into LLMs as a token representation within the input sequence.

4.3 Fallacy Tfaining

The LLMs take the instruction prompt, textualized ttee h1, and free-based soft prompt d6 as input, and generate fallacy label as output. The loss is calculated between the generated text and golden label. The text embedding layer and self attention layers

of LLMs are frozen. The ffee-based soft prompt d1 receives gradients and enables back propagation.

5 Experiments

5.1 Datasets

We experiment with four datasets from various domains and gemes. Table 3 shows their statistics. Argotario (Habernal et al,ZAl7) collects fallacies from the general domain question-answering pairs. The dataset includes the following fallacy labels: Ad Hominem, Appeal to Emotion, Hasty General- ization, Irrelevant Authority, Red Herring, and No Fallacy. We use this dataset for both fallacy detec- tion and classiflcation experiments, and follow the dataset splitting method in Alhindi et al. (2022).

Reddit (Sahai et a1.,2021) collects user generated posts from Reddit, and annotates logical fallacies into: Slippery Slope, Irrelevant Authority, Hasty Generalization, Black- and-White Fallacy, Ad Popu- lum, Tradition Fallacy, Naturalistic Fallacy, Worse Problem Fallacy, and No Fallacy. This dataset is used for both fallacy detection and classification.

Climate (Alhindi et a1.,2022) collects statemenrs from articles in the climate change domain, and annotated the following fallacies: Evading the Burden of Proof, Cherry Picking, Red Herring, Strawman, Irrelev ant Authority, H asty G eneralization, Fals e Cause, False Analogy, Vagueness, and No Fallacy.

Logic (Jin et a7.,2022) annotates logical fallacies in the educational materials into 13 types includ ing Ad Hominem, Ad populum, False Dilemma, False Cause, Circular Reasoning, Deductive Fal- lacy, Appeal to Emotion, Equivocation, Fallacy of Extension, Faulty Generalization,Intentionoiroi- lacy, Fallacy of Credibility, Fallacy of Relevance. This dataset does not include No Fallacy class and is only used for fallacy classification.

5.2 Experimental Settings

To validate our approach, we experiment on two types oflanguage models: a decoder-only model and an encoder-decoder model. For the decoder- only model, we choose the open-source large lan- guagemodelLlama-Z(llama-2-7b-chat-hf)(Tou- vron et a1.,2023). For the encoder-decoder model, we choose the Flan-T5-large model (Chung et al., 2022). Both the models are trained in a generative setting, where they take the instruction and given text as input, and generate a fallacy label as output. The fallacy detection task generates "Yes" or "No" label as output, while the fallacy classification task generates the name of each fallacy type. We follow Alhindi et aL. (2022) to unify the different names of the same fallacy across datasets, such as Fclse Dilemma is converted into Black-and-White Fal- lacy since they are the same fallacy. We also follow Alhindi et al. (2022) to feed the definitions of each fallacy type into the instruction prompt, The details of instruction prompt are explained in Appendix B. The maximum input length is set to be 1024, number of epochs is 10, weight decay is 1e-2, the gradient accumulation step is 4, learning rate for Llama-Z is 3e-4, and learning rate forFlan-T5 is 3e-5. The Llama-2 model is trained with LoRA (Hu et a1.,2021), with rank 8, alpha 16, dropout 0.05, and trainable modules include $q_projandv_proj$.

5.3 Baselines

We compare our models with the baselines listed below. Besides the existing baselines, we also im- plement several additional baselines based on the GPT and RoBERTa (Liu et al"2019) models: Sahai et al. (2021): a multi-granularity network is designed that trains sentence-level representation and the token-level representationsjointly. Jin et al. (2022): a structure-aware framework is de- veloped that forms a sequence-based logical pattern for each text by masking out the content words. Sourati et aJ. (2023b): a prototype-based reason- ing method that injects background knowledge and explainable mechanisms into the language model. Sourati et al. (2023a): a case-based reasoning that retrieves similar cases from external sources based on goals, counterarguments, and explanation etc. Alhindi etat. (2022)ta mutti-task instruction tun- ing framework that rearns the rogicar falacies frJm multiple datasets collaboratively. GPT-3.5: we prompt the gpt-3.5-turbo model to automatically choose one of the fallacy labels for each text, and the prompt is listed in Appendix C. GPT-3.5 * Ttogtc: guide the gpt-3.5-turbo model to firstly reason the logical structure of each text, and then choose one of the fallacy labels through a chain-of+hought process (Wei et a1.,2023). RoBERTa: the RoBERra model

is used to encode the texr and the average of word embedding is used as rhe text embedding. A classification head is built on top of the text embedding to classify labels. RoBERTa * Ttostci we concatenate the text embed- ding with the logical structure tree embedding,.and build classification head on top of the combined embedding to predict labels. The tree embedding is derived based on the method in Section 4.2.

5.4 Fallacy Detection

The fallacy detection task identifies whether a given text contains logical fallacy or not, which is a binary classification task. The precision, recall, and Fl score of thefallacy class, as well as the micro Fl score (i.e., accuracy) are used as evaluation metrics. Table2 presents the performance on the Argotario, Reddit, and Climate datasets. The results demonstrate that incorporating the logical structure tree effectively improves both pre- cision and recall for logical fallacy detection. This observation is consistent for both types of Llama-2 and Flan-T5 models across all the three datasets, which span various domains and genres. Compared to the baselines without logical structure informa- tion, our approach based on logical structure tree notably enhances the precision and recall, leading to the Fl score increased by up to 3.45vo. This indicates that the logical structure tree is effective in capturing the difference in logical flows between fallacious and benign texts.

Moreover, informing the large language model GPT-3.5-turbo of logical structure information significantly improves fallacy detection under the zero-shot setting, resulting in a substantial improvement in the Fl score. This underscores the importance of incorporating the logical structure information into LLM for fallacy detection. Also, concatenating the logical structure tree embedding with the text embedding in the RoBERTa model also enhances the performance, which proves the usefulness of this logical structure tree embedding. Overall, incor- porating the logical structure tree helps improve fallacy detection for various types of models.

5.5 Fallacy Classification

The fallacy classification task classifles the fallacy types for the fallacious text, which is a multi-class classification task excluding the No Fallacy class. The macro precision, recall, and Fl score, as well as the micro Fl score (i.e., accuracy) are used as evaluation metrics. Table 4 shows the results on the Argotario, Reddit, and Logic datasets. The results demonstrate that integrating the logi- cal structure tree into Llana-2 and Flan-T5 models notably enhances the performance of fallacy classi- fication, with both precision and recall increased. This conclusion is valid across the three datasets from different domains and genres. Compared to the baselines without logical structure tree, our pro- posed approach signifi cantly improves precision and recall, leading to an increase of up to 6.75Vo in the Fl score. This suggests that the logical structure tree effectively distinguishes the different logical patterns used in each fallacy type, and is applicable across various domains and genres.

In addition, our approach based on the logical structure tree outperforms the previous methods that may lack logical relations information. This highlights the necessity to infuse the logical rela- tions into LLMs for fallacy classification. Besides, our approach achieves higher performance than the baselines that overlook content words. This indicates that analyzing content words also plays an essential role in fallacy reasoning. The logical structure tree connects the logical relations and con- tent arguments together to form a cohesive logical structure, representing the hierarchical logical flow and thereby improving fallacy classification.

5.6 Ablation Study

The ablation study of the two designed strategies to incorporate the logical structure tree into LLMs is shown in Table 5, where we take Llama-2 model as an example. The upper rows show the results of fallacy detection on the three datasets, and the lower rows show the results of fallacy classification. The results demonstrate that both the textualized tree and tree-based soft prompt brings improvement for fallacy detection and classification across mul- tiple datasets. This proves that the textualized tree and tree-based soft prompt are complementary with each other: the

textualized tree enriches the instruc- tion prompt with logical structure information, and the tree-based soft prompt enables direct learning from the tree embedding. Comparing across these two strategies, the soft prompt usually achieves bet- ter performance than the hard text prompt, and ex- hibits higher recall. Combining the two strategies together leads to the best performance, achieving the highest precision and recall.

5.7 Effect on Different Fallacy Tlpes

We further analyze the F1 score change across each fallacy type in the fallacy classification task. The Llama-Z model is used as an example to show the performance change before and after incorporat- ing the logical structure tree. Table 6 presents the F1 score change across each fallacy type on Ar- gotario dataset. The performance change across each fallacy type on the Reddit and Logic dataset are shown in the Table 7 and Table 8. We ob- serve that the logical structure tree brings bigger improvements for the fallacy types such as Red Herring, Hasty Gerueralization, Irrelev ant Author- ity, Ad Populum, Extension Fallacy, Equivocation, Circular Reasoning etc. One possible explanation is that these fallacy types usually employ certain logical relations or logical patterns to persuade the readers. However, the performance increase is less noticeable for the fallacy types such as Appeal to Emotion and Ad Hominem. Itmay due to the reason that these fallacies rely more on the emotional or sentimental language instead of logical relations.

6 Limitations

We have compiled a set of connective words and phrases for the ten logical relations, as detailed in Table 1. While we have included the common connectives in this set, it may not contain all the possible connectives. The logical structure tree that is constructed based on this connective words set demonstrates its usefulness in fallacy reasoning. Future work can be expanding this connectives set and investigating the effects of various connectives, so that we can better identify and mitigate them. The release of code, datasets, and model should be used for mitigating logical fallacies, instead of expanding or disseminating the misinformation.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable feedback and input. We gratefully acknowledge support from National Science Foundation via the award trS2127746. Portions of this research were conducted with the advanced com- puting resources provided by Texas AM High- Performance Research Computing.

7 Conclusion

This paper detects and classifles fallacies. We propose a logical structure tree to explicitly represent and track the hierarchical logic flow among relation connectives and their arguments. we also design two strategies to incorporate this logical structure tree into LLMs for fallacy reasoning. Extensive experiments demonstrate the effectiveness of our approach based on the logical structure tree.

Ethical considerations

rhis paper aims to detect and classify logicat fallacies. Logical fallacy is the error or flaws in the reasoning, and can occur in various human com- munications. Logical fallacies can lead to harmful consequences ror soclety' sucn as spreaomg mls- information or introducing societal bias' The goal of this research is to understand logical fallacies,

References

Rehab Mohamed Ahmed Abd-Eldayem .2023. Therelationship between cognitive bias and logical fallacies in egyptian society' Social Sciences, 12(6):281-293,

Esma Aimeur, Sabrine Amri, and Gi]les Brassard. 2023. Fake news, disinformation and misinformation in social media: areview. SocialNetworkAnalysis and Mining, 13(1):30'

Tariq Alhindi, Tuhin Chakrabarry, Elena Musi, and Smaranda Muresan' 2022' Multitask instruction-based prompting for fallacy recognition. ln Proceedi:r;:,t!;';"';";:f"i:;r':f;::ti;::, Abu Dhabi, Unired Arab Emiritei. Association for Computational Linguistics.

Rachel Armitase and Cristian Vaccari. 2021. Misinformation andiisinformation. In The Routledge com-

panion to rnedia disinformation and populisi,pages 38-48. Routledge.

Donald A Barclay. 2018. Fake news, propaganda, and plain old lies: how to find trustworthy information in the digital age. Rowman Littlef,eld.

Sven Beisecker, Christian Schlereth, and Sebastian Hein. 2024. Shades offake news: How fallacies influence consumers' perception. European Journal of Infor- mation Systems, 33(1):41-60.

Claire Bonial, Austin Blodgett, Taylor Hudson, Stephanie M. Lukin, Jeffrey Micher, Douglas Summers-Stay, Peter Sutor, and Clare Voss. 2022. The search for agreement on logical fallacy annota- tion of an infodemic. In Proceedings of the Thir- teenth Language Resources and Evaluation Confer- ence, pages 44304438, Marseille, France. European Language Resources Association.

Zeming Chen, Qiyue Gao, and Lawrence S. Moss. 2021. Neurallog: Natural language inference with joint neural and logical reasoning. In Proceedings of *SEM 2021: The Tenth loint Conference on Lexical and Computational Semantics, pages 78-88, Online. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Banet Zoph,Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdh- ery AIex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Ja- cob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wel 2022. Scaling instruction-flnetuned language models. P rep rint, arXiv :2210. 1 1 4 I 6.

Christian Cotton. 2018. Argument from fallacy. Bad arguments: 100 of the most important fallacies in Western philosophy, pages 125-127 .

Antonia Creswell, Murray Shanahan, and Irina Higgins. 2022. Selection-inference: Exploiting large language models for interpretable logical reasoning. Preprint, arXiv:2205.09712.

Giovanni Da San Martino, Alberto Barr6n-Cedeflo, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020. SemEval-2020 task I l: Detection of propaganda techniques in news articles. In Proceedings of the Fourteenth Workshop on Semantic Evaluation, pages 1377-1414, Barcelona (online). Intemational Committee for Computational Linguistics.

Giovanni Da San Martino, Seunghak Yu, Alberto Ban6n-Cedeflo, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis ofpropaganda in news article. In Proceedings of the 2019 Conference on Empirical Methods in Natural lnnguage Processing and the 9th International Joint Conference on Natural Lan g u a g e P ro c e s s in g (EMN LP - I J C N LP.), pages 5636-5646, Hong Kong, China. Association for Com- putational Linguistics.

Nicolas Devatine, Philippe Muller, and Chlo6 Braud. 2023. An integrated approach for political bias pre- diction and explanation based on discursive structure. Irr Findings of the Association for Computational Lin- guistics : ACL 202 3, pages I 1 196-1 12 1 1, Toronto, Canada. Association for Computational Linguistics.

Edmund Fantino, Stephanie Stolarz-Fantino, and An- ton Navarro. 2003. Logical fallacies: A behavioral approach to reasoning, The B ehavio r Analys t To day, 4(1):109,

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. 2023. From pretraining data to language ' models to downstream tasks: Tracking the trails of political biases leading to unfair NLP models. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volurne I : Long Papers), pages 11737-11762, Toronto, Canada. Association for Computational Linguistics.

Pierpaolo Goffredo, Mariana Chaves, Serena Villata, and Elena Cabrio, 2023. Argument-based detection and classiflcation of fallacies in political debates. In Proceedings of the 2023 Conference on Empir- ical Methods in Natural Language Processing, pages 1 1 10 1-1 1 1 1 2, Singapore. Association for Computational Linguistics.

Pierpaolo Goffredo, Shokeh Haddadan, Vorakit Vorak- itphan, Elena Cabrio, and Serena Villata. 2022. Fal- lacious argument classification in political debates. kt IJ CN, pages 41434149.

William S Greenwell, John C Knight, C Michael Hol- loway, and Jacob J Pease. 2006. A taxonomy of fallacies in system safety argumerfis. In24th Intemational System Safety Conference.

Andrew M Guess and Benjamin A Lyons. 2020. Mis- information, disinformation, and online propaganda. Social media and dernocracy: The state of the field, prospects for reform, 10.

Ivan Habernal, Raffael Hannemann, Christian Pollak, Christopher Klamm, Patrick Pauli, and Iryna Gurevych. 20L1 . Argotario: Computational argumentation meets serious games. In Proceedings of the 2017 Conference on Empirical Methods in Nat- ural l,an gua g e P roc e s s in g : Sy s t em D emon s tr ati ons, pages 7 -12, Copenhagen, Denmark, Association for Computational Linguistics,

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. Before name-calling: Dynamics and triggers of ad hominem fallacies in web argumentation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Lan- g ua g e Te c hno l o g i e s, Vo lume I (Lo n g Pap e r s), p ages 386-396, New Orleans, Louisiana. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeytan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen.202l. Lora: Low-rank adaptation of large language models. P reprint, arXiv [ILLEGIBLE]

Zh\itg Jin, Abhinav Lalwani, Tejas Vaidhya, Xiaoyu Shen, Yiwen Ding, Zhiheng Lyu, Mrinmaya Sachan, Rada M [ILLEGIBLE]

Fangkai Jiao, Yangyang Guo, Xuemeng Song, and Liqiang Nie.2022. MERIT: Meta-Path Guided Con- trastive Learning for Logical Reasoning. In Find- ings of the Association for Computational Linguistic s : AC L 2 022, pages 349 6-3 509, Dublin, Ireland. Association for Computational Linguistics.

[ILLEGIBLE]

Elena Musi and Chris Reed.2022. From fallacies to semi-fake news: Improving the identification of mis- information triggers across digital media. Discourse S o ciety, 33 (3):3 49-37 0.

Callistus Ireneous Nakpih and Simone Santini. 2020. Automated discovery of logical fallacies in legal ar- gumentation. International Journal of Artificial In- telligence and Applications (IJNA), ll.

Theo Olausso [ILLEGIBLE]

Christopher W Tindale. 2007. Fallacies and argument appraisal. Cambridge University Press.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Al- bert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu,

TABLE CONTENT [ILLEGIBLE]

Table 2: Table 9: The ratio (Vo) of samples that contain the ten logical relations in fallacy and, no fallacy classes in the development set of Argo

TABLE CONTENT [ILLEGIBLE]

Table 3: Table 10: The ratio (Vo) of samples that contain the ten logical relations in each fallacy type in the development set of Argo

Cynt [ILLEGIBLE]
[ILLEGIBLE]

A Statistical Analysis of Logical Relations C Prompt for GPT-based baselines

Table 9 presents the ratio of samples that contain the ten logical relations infallacy and no fallacy classes, where we take the Argotario (Habernal et a1., 2017) and Reddit (Sahai et al., 2021) datasets as examples. Further, Table 10 shows the ratio of samples that contain the ten logical relations in each fallacy type

B The Names and Definitions of Fallacies

B.1 Argotario dataset

The Argotario dataset (Habernal et a1.,2017) in- cludes five fallacy types: Ad Hominem, Appeal to Emotion, Hasty Generalization, Irrelevant Author- ity, Red Herring. The name of Appeal to Emotion is converted into Emotional Language. The defini- tions of these fallacy types which are used in the instruction prompt are:

. Ad Hominem: [MISSING]
. Emotional Language: [MISSING]
. Hasty Generalization: [MISSING]
. Irrelevant Authority: [MISSING]
. Red Herring: [MISSING]

B.2 Reddit dataset

The Reddit dataset (Sahai et a1.,2021) includes eight fallacy types and one no fallacy type. The fallacy types include: Slippery Slope, Irrelevant Authority, Hasty Generalization, Black-and-White Fallacy, Ad Popu- lum, Tradition Fallacy, Naturalistic Fallacy, Worse Problem Fallacy. The definitions of these fallacy types which are used in the instruction prompt are: [ILLEGIBLE]

B.3 Climate dataset

The Climate dataset (Alhindi et a1.,2022) includes nine fallacy types and one no fallacy type. The fallacy types include: Evading the Burden of Proof, Cherry Picking, Red Herring, Strawman, Irrelevant Authority, Hasty Generalization, False Cause, False Analogy, Vagueness. The definitions of these fallacy types which are used in the instruction prompt are: [ILLEGIBLE]

B.4 Logic dataset

The Logic dataset (Jin et a7.,2022) includes 13 fallacy types: Ad Hominem, Ad Populum, False Dilemma (Black-and-White Fallacy), False Cause, Circular Reasoning, Deductive Fallacy, Appeal to Emotion (Emotional Language), Equivocation, Fallacy of Ex- tension, Faulty Generalization (Hasty Generalization), Intentional Fallacy, Fallacy of Credibility (Irelevant Au- thority), Fallacy of Relevance (Red Hening). The names in the parenthesis are the replaced names used in the instruction prompt. The definitions of these fallacy types which are used in the instruction prompt are: . Ad Hominem: the text attack a person instead of arguing against the claims. . Ad Populum: the text affirm something is true

because the majority thinks so, . Black-and-White Fallacy: the text present two alternativ [ILLEGIBLE]