Improved Logical Reasoning of Language Models via Differentiable Symbolic Programming

Anonymous Authors[1]

Abstract

Logical reasoning is a long-standing fundamental challenge in AI. Even pre-trained large language models (LMs) with hundreds of millions of parameters and improved compositionality struggle to perform logical reasoning reliably. In this work, we tackle this challenge through the lens of symbolic programming. We propose Salmon, a neuro-symbolic framework where pre-trained LMs govern the perception of factual knowledge and a provenance symbolic model generates top-k proofs by deductive reasoning based on the given logical rules. The symbolic model is scalable, interpretable, and allows to easily integrate prior knowledge, thereby supporting extensive symbolic programming to robustly derive a logical conclusion. We conduct experiments to show that Salmon leads to improved logical reasoning in pre-trained LMs and consistently outperforms a spectrum of competitive baselines even under systematic distribution shifts on sequence lengths.

1. Introduction

Pre-trained LMs have demonstrated remarkable predictive performance but they tend to generate outputs that are harmful, inconsistent, untruthful, toxic, or reflect harmful sentiments (Weidinger et al., 2021), in part because their systematic understanding and reasoning capability about factual knowledge is severely lacking (Elazar et al., 2021; Hase et al., 2021). As one of the most notable cases, it has been long observed that logical inconsistency is one of the significant weaknesses of LMs (Kassner et al., 2020; Helwe et al., 2021; Creswell et al., 2022), even in a confined problem space (Zhang et al., 2022). For example, LMs cannot predict consistently the answer fly given the queries with the same meaning: "Birds can [MASK]" and "Birds are able to [MASK]"; LMs also tend to fail catastrophically with logical relationships like rules includng negation "(giraffe, is, tall) $\Longleftrightarrow$ (giraffe, is not, short) etc (Kassner et al., 2020). Such observations are notable evidence that pre-trained LMs are far from human-level understanding.

In this work, we seek to tackle the above challenges and focus on improving the deductive logical reasoning process of LMs. We first revisit some key desiderata of neuro-symbolic programming (Chaudhuri et al., 2021): compositionality, consistency, interpretability and easy integration of prior knowledge. Although many attempts have been made in neuro-symbolic reasoning (Rocktäschel & Riedel, 2017; Manhaeve et al., 2018; Zhang et al., 2019), we argue that none of them are suitable to improve pre-trained LMs since they are designed to be self-consistent in a few confined problem spaces. To this end, we develop Salmon, which supports a tight integration of pre-trained LMs and symbolic programming to leverage the advantages of both worlds. Specifically, Salmon integrates LMs with symbolic programmers (Huang et al., 2021) in a principled manner: the LM governs the perception of `(relation, subject, object)` tuples and the symbolic programmer efficiently and deductively reasons with desirable encoded logical rules to generate the top-k proofs via forward chaining for predicting the truth of a query tuple. The overall process can be illustrated with a motivating example Figure 1.

We conduct extensive experiments showing that Salmon can consistently improve the logical reasoning capability or pre-trained LMs. As generalization over long-range dependency is a significant weakness of transformer-based language models (Tay et al., 2020), we highlight that in systematic, long-context scenarios, where most pre-trained or neural approaches fail to generalize compositionally, our model can still achieve considerable performance gains.

2. Related Works

Logical Reasoning with LMs. Recent works show that Transformers can perform deductive reasoning tasks when finetuned over synthetic datasets (Clark et al., 2021). However, negative results are shown pre-trained LMs are not good (logical) reasoners over factual knowledge (Kass-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the First Workshop of Pre-training: Perspectives, Pitfalls, and Paths Forward at ICML 2022. Do not distribute.

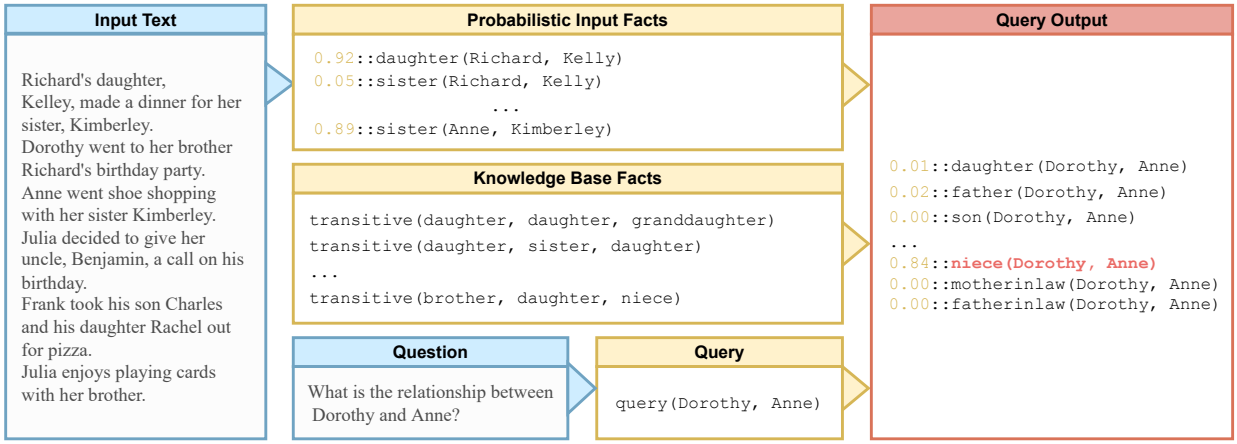

Figure 1. A motivating example where "Anne is the niece of Dorothy" should be inferred from the context.

ner et al., 2020; Helwe et al., 2021; Talmor et al., 2020a). Though there are some encouraging progress (Clark et al., 2021; Wei et al., 2022; Chowdhery et al., 2022; Zelikman et al., 2022), they usually require a significant amount of computation for re-training and human annotations on reasoning paths or so-called chain of thoughts (Wei et al., 2022). Moreover, their entangled nature with natural language make them fundamentally hard to make robust inference over symbolic factual knowledge (Greff et al., 2020). There are other obvious remedies for LMs' poor reasoning capability: make sure that the training corpus contains a sufficient amount of exemplary episodes of sound reasoning to reduce the dependency on normative biases and annotation artifacts (Talmor et al., 2020b; Betz et al., 2020; Hase et al., 2021). Heuristics like data augmentation is also shown to be effective (Talmor et al., 2020b). But the above works require significant efforts for crowdsourcing and auditing training data. Our method handily encodes a few prototypes/templates of logical rules, which is more efficient in human efforts. However, our goal is fundamentally different from theirs in investigating the tight integration of neural and symbolic models in an end-to-end manners.

Neuro-Symbolic Reasoning. Neuro-symbolic approaches are proposed to integrate the perception of deep models and the reasoning of symbolic models. Representative works can be briefly categorized into regularization (Xu et al., 2018), program synthesis (Mao et al., 2018), proof-guided probabilistic programming (Rocktäschel & Riedel, 2017; Manhaeve et al., 2018; Zhang et al., 2019; Huang et al., 2021).

3. Method

Problem formulation. Each data-point is a tuple of 3 elements, namely (context, query, answer). The context is a set of natural language sentences that entail a story, within which will be a set of people names, possibly referenced by third person pronouns. These sentences will hint the relationships between the people, while the name of these people might also suggest their gender. For example, "Diane is taking her brother Brad out for a late dinner" implies that *Brad* is *Diane*'s brother and *Diane* is *Brad*'s sister. The query, on the other hand, is a tuple of two names, denoting the relationship between whom we want to infer. The expected relation is stored in the answer, which will be one of 20 basic relations such as "mother", "uncle", and "daughter-in-law".

We treat the whole problem as a 20-way classification problem where exactly one relation among the 20 will be the answer. There are a few major difficulties of this problem:

1. The way relationships are expressed can vary a lot, indicating a need for large LM to understand the meaning of these phrases. For example, one might use "John, the father of the father of Alice, is..." to denote that John is the grandfather of Alice.
2. To predict the relationship between the two people in the desired query, multi-hop reasoning is needed. In fact, the length of the reasoning chain is denoted by a parameter k. Usually, the larger the k is, the longer the context will be, making the reasoning even harder.
3. Lastly, to test the ability for systematic generalization, we train on contexts with $k = 2$ and $k = 3$ while we test our model on longer contexts upto $k = 10$.
4. During training, we do not obtain supervision for any intermediate relation, but only the final answer relation for the query. This is suggesting that we have a weak-supervision learning setup.

Methodology Overview. To overcome the difficulties listed above, we make the following design decisions. First, we employ and fine-tune large LMs such as RoBERTa to

perform the basic relation prediction over shorter windowed contexts. This way, we can ensure that LMs are reasoning about similar-length-context when k becomes larger during test time. Next, we will turn the predicted relations into a probabilistic database, before feeding it into a differentiable logic reasoning engine, Scallop (Huang et al., 2021). Lastly, Scallop will perform multi-hop reasoning that yields a predicted distribution, which is passed to a binary-cross-entropy (BCE) loss at the end. Since the logical reasoning module is fully differentiable, this is an end-to-end training loop that we can back-propagate gradients from the loss function to fine-tune our LMs.

Relation Extraction. The goal of relation extraction is to obtain a distribution of relations between every pair of family members in each windowed context. There are in total 20 base relations that we want to reason about, including mother, son, uncle, father-in-law, and etc. We first assume there is a perfect Named-Entity-Recognition (NER) system that can extract family members' names as entities. From here, for each pair of names, we will construct a feature vector for relation classification. Such feature vector for a pair of names A and B is obtained by concatenating the aggregated embeddings for names A and B and the whole windowed context embedding. Lastly, we apply an MLP classifier over the feature vector to obtain the distribution of relations between this pair of names.

Note that the pair-wise names are directional, i.e. the relation between *Diane* and *Brad* is not the same as the relation between *Brad* and *Diane*. Therefore, if a windowed context contains n distinct names, we will predict $n(n-1)$ distributions of relation. Additionally, for a given pair of name there might not be any implied relation. Therefore we employ a 21-way classification where the additional class represents "no relation" between the two names.

Neuro-Symbolic Reasoning. The neural symbolic reasoning model will combine the information from multiple windowed texts, and deduce the answer for the multi-hop reasoning required query. The extracted relations form a probabilistic database containing context and query. The context relation is modeled as a probabilistic arity-3 predicate `context(relation, subject, object)`. For example, `context(son, Alice, Bob)` means "Bob is Alice's son". For each pair of names, the 21 possible context relations form a mutually exclusive disjunction – that is, if A is B's father, then A cannot be B's mother or any other relation. This can prevent the reasoning process from generating dubious proof that can confuse the prediction.

For this CLUTRR task, it suffices to define a higher-order predicate called *transitivity*. The transitivity defines how two known relations can be connected together to derive the third relation. For example, the rule "If B is A's son, and C is B's daughter, then C is A's granddaughter" can be represented as a transitivity fact `transitive(son, daughter, granddaughter)`. We treat these transitivity facts as common-sense knowledge and injected 92 such facts into our program. At the end, we use the query `query(subject, object)` to inspect the derived answer relation. The full Scallop program used for this is shown in Figure 2.

4. Experiments

The goal of our experiments is to compare the performance of Salmon over a wide range of baselines in challenging scenarios involving long-context, systematic generalization.

Settings. We evaluate Salmon on the CLUTRR datasets. CLUTRR (Sinha et al., 2019) concerns a question answering tasks that involve logic programming over knowledge bases: given a group of knowledge graphs, where each node denotes a family member and edges are family relations, the goal of **CLUTRR** task is to infer two-family members' relationship that is not explicitly mentioned. The training set of CLUTRR consists of graphs that the target relation can be inferred by traversing a limited number of edges while the relation in the test set needs more traversing steps for inference.

Baselines. We compare Salmon with a spectrum of baselines including pretrained models (BERT (Kenton & Toutanova, 2019), RoBERTa (Liu et al., 2019)), structured models (GAT (Veličković et al., 2018), RN (Santoro et al., 2017), MAC (Hudson & Manning, 2018)), non-LM counterparts (BiLSTM (Hochreiter & Schmidhuber, 1997; Cho et al., 2014) and BERT-LSTM). Structured models include those models with relational inductive biases like Graph Attention Networks, Relational Networks (RN), Compositional Memory Attention Network (MAC). BERT-LSTM is a trainable LSTM encoder taking pretrained BERT embeddings as inputs.

Empirical studies on controllable settings. The overall accuracy comparison on CLUTRR is summarized in Figure 3, where a significant performance gain of Salmon can be observed. We also include a full set of experimental results in Figure 4. All models are trained over sequences with reasoning chains of length 2 while tested on those with length $k \in [2, 10]$. Overall, all models' performance decline as the length of test sequence increases, which aligns with previous observations that sequence models are not robust under compositional or systematic distribution shifts (Lake & Baroni, 2018). Note that even with relational inductive biases, GAT still can not reliably generalize beyond the training distribution. In fact, with Scallop, even a non-LM model like Word2Vec + BiLSTM improved with Scallop can beat most models. Although the training of our method (Scallop + RoBERTa) achieves its highest performance in

```
// Base facts
type context(rela: usize, subject: String, object: String)
type query(subject: String, object: String)
type transitive(r1: usize, r2: usize, r3: usize)
rel transitive = { /* 92 transitivity facts */ }

// Rules to derive the final answer
rel derive(r, s, o) = context(r, s, o)
rel derive(r3, x, z) = derive(r1, x, y), derive(r2, y, z), x != z, transitive(r1, r2, r3)
rel answer(r) = query(s, o), derive(r, s, o)
```

Figure 2. The Scallop program used for kinship reasoning

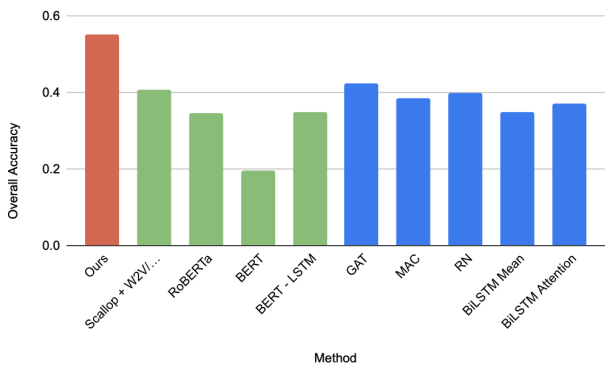

Figure 3. Overall performance.

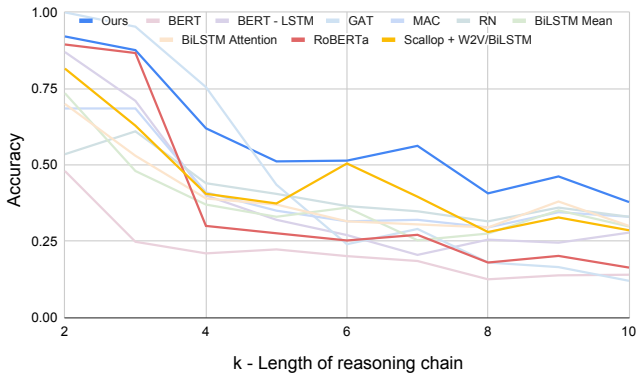

Figure 4. Systematic generalization evaluation. Trained on $k = 2$ and $k = 3$ and test on length $k \in [2, 10]$.

just 4 epochs, while Scallop + W2V/BiLSTM takes 30.

Ablation study. We conduct ablation study for answering the following questions: (i) How does symbolic programming improve upon pre-trained LMs? (ii) How does symbolic programming works only with non-LM counterparts? Figure 4 suggests that Salmon outperforms both ablation baselines (RoBERTa and Scallop+W2V/BiLSM) consistently. As test sequence length increases, RoBERTa far lags behind the two neuro-symbolic counterparts.

We also conduct ablation study to probe the effects of weak supervision in long-context reasoning. Specifically, we include a baseline that trains RoBERTa to predict all the relationships of intermediate steps. It is shown that it can achieve 98.46% training accuracy and 85.62% testing accuracy on single pair-wise relation prediction. We then use it to extract the top-1 relation for each pair of family members followed by discrete logical reasoning to infer the relation of the desired query, and the end accuracy is 53.25%. This suggests that our model trained with weak supervision can stay on-par and even exceed the model trained with full intermediate supervision.

Failure case analysis. We showcase in Appendix Table 1 that even state-of-the-art large LMs are prone to logical fallacies. On the other hand, the failure case of our method usually occur in the stage of relation extraction. For example, for the following sentence "Christopher and Guillermina are having a father-daughter dance", our relation extractor fails to recognize the father-daughter relationship but rather thinks Christopher and Guillermina have a husband-and-wife. This error cascades and caused the derivation of a wrong answer.

5. Concluding Remarks

In this work, we investigate how to improve LMs' logical reasoning capability using differentiable symbolic programming. Extensive experiments well verify the effectiveness and utility of Salmon over challenging scenarios where widely deployed large LMs fail to reason robustly. We hope our work can lay the groundwork for exploring techniques from neuro-symbolic programming to improve the robustness of LMs. In the future, we plan to improve LMs' reasoning over complex compositional scenarios like natural language satisfiability (NLSat) problems (Richardson & Sabharwal, 2021). We also leave for future work using temporal logic to model dynamic semantics and *state tracking* in dialog systems - identifying the user's intent and parameters in each dialog act, an use them to drive the system's actions consistently (Andreas et al., 2020).

References

Andreas, J., Bufe, J., Burkett, D., Chen, C., Clausman, J., Crawford, J., Crim, K., DeLoach, J., Dorner, L., Eisner, J., et al. Task-oriented dialogue as dataflow synthesis. *Transactions of the Association for Computational Linguistics*, 8:556–571, 2020.

Betz, G., Voigt, C., and Richardson, K. Critical thinking for language models. *arXiv preprint arXiv:2009.07185*, 2020.

Chaudhuri, S., Ellis, K., Polozov, O., Singh, R., Solar-Lezama, A., Yue, Y., et al. *Neurosymbolic Programming*. Now Publishers, 2021.

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Clark, P., Tafjord, O., and Richardson, K. Transformers as soft reasoners over language. In *IJCAI*, 2021.

Creswell, A., Shanahan, M., and Higgins, I. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv preprint arXiv:2205.09712*, 2022.

Elazar, Y., Kassner, N., Ravfogel, S., Ravichander, A., Hovy, E., Schütze, H., and Goldberg, Y. Measuring and improving consistency in pretrained language models. *TACL*, 2021.

Greff, K., Van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Hase, P., Diab, M., Celikyilmaz, A., Li, X., Kozareva, Z., Stoyanov, V., Bansal, M., and Iyer, S. Do language models have beliefs? methods for detecting, updating, and visualizing model beliefs. *arXiv preprint arXiv:2111.13654*, 2021.

Helwe, C., Clavel, C., and Suchanek, F. M. Reasoning with transformer-based models: Deep learning, but shallow reasoning. In *AKBC*, 2021.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 1997.

Huang, J., Li, Z., Chen, B., Samel, K., Naik, M., Song, L., and Si, X. Scallop: From probabilistic deductive databases to scalable differentiable reasoning. *NeurIPS*, 2021.

Hudson, D. A. and Manning, C. D. Compositional attention networks for machine reasoning. In *ICLR*, 2018.

Kassner, N., Krojer, B., and Schütze, H. Are pretrained language models symbolic reasoners over knowledge? *arXiv preprint arXiv:2006.10413*, 2020.

Kenton, J. D. M.-W. C. and Toutanova, L. K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners, 2022.

Lake, B. and Baroni, M. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *ICML*, 2018.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., and De Raedt, L. Deepproblog: Neural probabilistic logic programming. *NeurIPS*, 2018.

Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B., and Wu, J. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *ICLR*, 2018.

Richardson, K. and Sabharwal, A. Pushing the limits of rule reasoning in transformers through natural language satisfiability. *arXiv preprint arXiv:2112.09054*, 2021.

Rocktäschel, T. and Riedel, S. End-to-end differentiable proving. *NeurIPS*, 2017.

Santoro, A., Raposo, D., Barrett, D. G., Malinowski, M., Pascanu, R., Battaglia, P., and Lillicrap, T. A simple neural network module for relational reasoning. *NeurIPS*, 2017.

Sinha, K., Sodhani, S., Dong, J., Pineau, J., and Hamilton, W. L. Clutrr: A diagnostic benchmark for inductive reasoning from text. *EMNLP*, 2019.

Talmor, A., Elazar, Y., Goldberg, Y., and Berant, J. olmpics-on what language model pre-training captures. *TACL*, 2020a.

Talmor, A., Tafjord, O., Clark, P., Goldberg, Y., and Berant, J. Leap-of-thought: Teaching pre-trained models to systematically reason over implicit knowledge. *NeurIPS*, 2020b.

Tay, Y., Dehghani, M., Abnar, S., Shen, Y., Bahri, D., Pham, P., Rao, J., Yang, L., Ruder, S., and Metzler, D. Long range arena: A benchmark for efficient transformers. In *International Conference on Learning Representations*, 2020.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *ICLR*, 2018.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., and Zhou, D. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, M., Balle, B., Kasirzadeh, A., et al. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*, 2021.

Xu, J., Zhang, Z., Friedman, T., Liang, Y., and Broeck, G. A semantic loss function for deep learning with symbolic knowledge. In *ICML*, 2018.

Zelikman, E., Wu, Y., and Goodman, N. D. Star: Bootstrapping reasoning with reasoning. *arXiv preprint arXiv:2203.14465*, 2022.

Zhang, H., Li, L. H., Meng, T., Chang, K.-W., and Broeck, G. V. d. On the paradox of learning to reason from data, 2022.

Zhang, Y., Chen, X., Yang, Y., Ramamurthy, A., Li, B., Qi, Y., and Song, L. Efficient probabilistic logic reasoning with graph neural networks. In *ICLR*, 2019.

A. Additional Experimental Results

In Table 1, we showcase the failure cases of large LMs for logical inference, where Zero-shot-CoT denotes zero-shot chain-of-thoughts (Kojima et al., 2022).

Method	Prompt	Output
Zero-shot-CoT	Harry and his sister Morgan went out for ice cream. kevin's daughter, Valerie, made a dinner for her sister, Melissa. Harry was having trouble with his divorce because he only got to see his daughter Isabel a few times a week. Melissa enjoyed a homemade dinner with her son Scott. Joy had picked her daughter Isabel out the cutest new dress to wear on her birthday. Scott and his Aunt Joy flew first class Danielle went to eat with her parents, Dale and Morgan. Danielle's sister, Ouida, wasn't able to join them. What is the relationship between Kevin and Ouida? Let's do it step by step:	1. Kevin's daughter, Valerie, made a dinner for her sister, Melissa. 2. Melissa enjoyed a homemade dinner with her son Scott. 3. Scott and his Aunt Joy flew first class. 4. Joy had picked her daughter Isabel out the cutest new dress to wear on her birthday. From these sentences, we can infer that Kevin is Melissa's father, Joy is Scott's aunt, and Isabel is Jov's daughter. Therefore, Kevin and Ouida are siblings.

Table 1. The comparison of different demonstrations on results.