based inductive logic programming system that requires manual annotation of data, such as mode declaration, finding group size, etc. Conversely, the SQuARE system relies fully on automatic reasoning with only manual encoding of *reusable* commonsense knowledge. An action language based QA methodology using VerbNet has been developed by Lierler et al (**?**). The project aims to extend frame semantics with ALM, *an action language* (**?**), to provide interpretable semantic annotations. Unlike SQuARE, it is not an end-to-end automated QA system.

Table 3 shows the accuracy of our proposal, StaCACK, against the best models on the bAbI dataset in terms of per-response and in parenthesis in terms of per-dialog: Mem2Seq (**?**), and BoSsNET (**?**). Other results can be found elsewhere (**?**). Unsurprisingly, similar to the rule-based system, StaCACK surpasses all the ML based models by showing 100% accuracy. Nevertheless, due to the commonsense reasoning, StaCACK can hold better natural conversation (shown in the *Example* section of *StaCACK*) that is not possible with a standard rule-based system based on monotonic logic.

Discussion

Our goal is to create NLU applications that mimics the way human understand natural language. Humans understand a passage's meaning and use commonsense knowledge to logically reason to find a response to a question. We believe that this is the most effective process to create an NLU application. *Learning* and *reasoning* both are integral parts of human intelligence. Today, ML research dominates AI. Most state-of-the-art CA or QA systems have been developed using ML techniques (e.g., LSTM, GRU, Attention Network, Transformers, etc.). The systems that are built with these techniques learn the patterns of the training text remarkably well and shows promising results on test data. With the recent advancements in the language model research, the pre-trained models such as BERT (**?**) and GPT-3 (**?**) have outstanding capability of generating natural languages. These rapid evolutions in the NLU world showcase some extremely sophisticated text predictor. These are used to build a chatbot or a QA system that can generate correct responses by exploiting the correlation among words and without properly understanding the content. These ML techniques are extremely powerful in tasks where learning of hidden data patterns is needed, such as machine translation, sentiment analysis, syntactic parsing, etc. However, they fail to generate proper responses where reasoning is required and they mostly do not employ commonsense knowledge. Also, the black-box nature of theses models makes their response non-explainable. In other words, these models do not possess any internal meaning representation of a sentence or a word and have no semantically-grounded model of the world. So, it will be an injustice to say that they understand their inputs and outputs in any meaningful way. Our semantic knowledge generation approach and its two applications are a step toward mimicking a human assistant. We believe that, to obtain truly intelligent behavior, ML and commonsense reasoning should work in tandem.

Compared to ML-based QA systems and CAs, our approach has many advantages. It produces correct responses by truly understanding the text and reasoning about it, rather than by using patterns learned from training examples. Also, ML-based systems are more likely to produce incorrect response, if not trained appropriately, resulting in vulnerability under adversarial attacks (**?**). We believe that our commonsense reasoning based systems are more resilient. Scalability is also an issue due to the dependence on training data. Explainability is a necessary feature that a truly intelligent system must possess. Both

Model / Tasks	MemNN (AM+NG+NL)	Mitra et al.	SQuARE
Single Supporting Fact	100	100	100
Two Supporting Facts	98	100	100
Three Supporting Facts	95	100	100
Two Arg. Relation	100	100	100
Three Arg. Relation	99	100	99.8
Yes/No Questions	100	100	100
Counting	97	100	100
Lists/Sets	97	100	100
Simple Negation	100	100	100
Indefinite Knowledge	98	100	98.2
Basic Coreference	100	100	100
Conjunction	100	100	100
Compound Coreference	100	100	100
Time Reasoning	100	100	100
Basic Deduction	100	100	100
Basic Induction	99	93.6	100
Positional Reasoning	60	100	100
Size Reasoning	95	100	100
Path Finding	35	100	100
Agents Motivations	100	100	100
MEAN ACCURACY	**94**	**100**	**100**

Table 2: SQuARE accuracy (%) comparison

	Mem2Seq	BossNet	StaCACK
Task 1	100 (100)	100 (100)	100 (100)
Task 2	100 (100)	100 (100)	100 (100)
Task 3	94.7 (62.1)	95.2 (63.8)	100 (100)
Task 4	100 (100)	100 (100)	100 (100)
Task 5	97.9 (69.6)	97.3 (65.6)	100 (100)
Task 1 (OOV)	94.0 (62.2)	100 (100)	100 (100)
Task 2 (OOV)	86.5 (12.4)	100 (100)	100 (100)
Task 3 (OOV)	90.3 (38.7)	95.7 (66.6)	100 (100)
Task 4 (OOV)	100 (100)	100 (100)	100 (100)
Task 5 (OOV)	84.5 (2.3)	91.7 (18.5)	100 (100)

Table 3: Accuracy per response (per dialog) in %.

Future Work and Conclusion

We presented our novel semantics-driven English text to answer set program generator. Also, we showed how commonsense reasoning coded in ASP can be leveraged to develop advanced NLU applications, such as SQuARE and StaCACK. We make use of the s(CASP) engine, a query-driven implementation of ASP, to perform reasoning while generating an natural language explanation for any computed answer. As part of fu-

ture work, we plan to extend the SQuARE system to handle more complex sentences and eventually handle complex stories. Our goal is also to develop an open-domain conversational AI chatbot based on automated commonsense reasoning that can "converse"