Utilizing Background Knowledge for Robust Reasoning over Traffic Situations

Jiarui Zhang[1], **Filip Ilievski**[1], **Aravinda Kollaa**[1],
Jonathan Francis[2], **Kaixin Ma**[3], **Alessandro Oltramari**[2]

[1]Information Sciences Institute, University of Southern California
[2]Human-Machine Collaboration, Bosch Research Pittsburgh
[3]Language Technologies Institute, Carnegie Mellon University
{jrzhang, ilievski, kollaa} @isi.edu, {jon.francis, alessandro.oltramari} @us.bosch.com, kaixinm@cs.cmu.edu

Abstract

Understanding novel situations in the traffic domain requires an intricate combination of domain-specific and causal commonsense knowledge. Prior work has provided sufficient perception-based modalities for traffic monitoring, in this paper, we focus on a complementary research aspect of Intelligent Transportation: traffic understanding. We scope our study to text-based methods and datasets given the abundant commonsense knowledge that can be extracted using language models from large corpus and knowledge graphs. We adopt three knowledge-driven approaches for zero-shot QA over traffic situations, based on prior natural language inference methods, commonsense models with knowledge graph self-supervision, and dense retriever-based models. We constructed two text-based multiple-choice question answering sets: BDD-QA for evaluating causal reasoning in the traffic domain and HDT-QA for measuring the possession of domain knowledge akin to human driving license tests. Among the methods, Unified-QA reaches the best performance on the BDD-QA dataset with the adaptation of multiple formats of question answers. Language models trained with inference information and commonsense knowledge are also good at predicting the cause and effect in the traffic domain but perform badly at answering human-driving QA sets. For such sets, DPR+Unified-QA performs the best due to its efficient knowledge extraction.

Introduction

As humans, we are able to make sense of novel situations in any domain even with limited domain-specific knowledge. The domain of intelligent traffic monitoring is a particularly attractive one, given the large market and the long list of car manufacturers that innovate in this domain (Int 2022). Being able to understand novel situations in traffic requires a complex association of domain-specific and causal commonsense knowledge. Prior work on evaluation tasks (Xu et al. 2017; Kim et al. 2018; Xu, Huang, and Liu 2021) and corresponding methods (Halilaj et al. 2021; Muppalla et al. 2017; Chowdhury et al. 2021; Wickramarachchi, Henson, and Sheth 2020) has focused on the perception-based modality, with many of the input exemplars coming from sensors like street video cameras.

Perception (e.g., tracking of participants over time, counting cars, etc.) is certainly a key aspect of traffic monitoring. However, a holistic understanding of situations in traffic requires other, complementary skills such as causal and commonsense reasoning (e.g., *red light causes cars to stop, direct sunlight does not*), and possession of rich domain knowledge (e.g., *the driving speed in school zones is limited to 30mph*). A large body of work has focused on general commonsense reasoning (Ma et al. 2021b; Zhang et al. 2022), but the role of commonsense and causal reasoning has not been explored thoroughly. Complementarily, ontologies for describing traffic participants (Ope 2020) exist, but it is unclear whether connecting them to neural models can bring the desired generalizability. Given that natural language is a common medium for human communication, and considering the success of large language models (LMs), it is intuitive to turn to using the textual modality to develop and test robust traffic understanding models. To the best of our knowledge, understanding traffic situations presented in natural language has not been explored in depth so far. This brings up a natural question of whether large pretrained models can effectively solve realistic traffic understanding tasks dominantly presented in language, such as human driving license tests. We also observe that prior work has not systematically investigated the role of different types of background knowledge, like causal commonsense knowledge and traffic domain knowledge, in reasoning over traffic situations.

In this paper, we focus on an evaluation for robust reasoning over traffic situations described in natural language. This allows us to reduce complexity, understand the behavior of various neural and neuro-symbolic models, and measure the impact of various sources of background knowledge. We construct two realistic text-based multiple-choice question answering (MCQA) sets: BDD-QA for evaluating causal reasoning in the traffic domain and HDT-QA for measuring the possession of domain knowledge akin to human driving license tests. We adopt three knowledge-driven approaches for zero-shot QA over traffic situations, based on prior methods for natural language inference, commonsense models with knowledge graph self-supervision, and dense retriever-based models. By doing so, we investigate the impact of ex-

Accepted to Workshop on Knowledge Augmented Methods for Natural Language Processing, in conjunction with AAAI 2023.

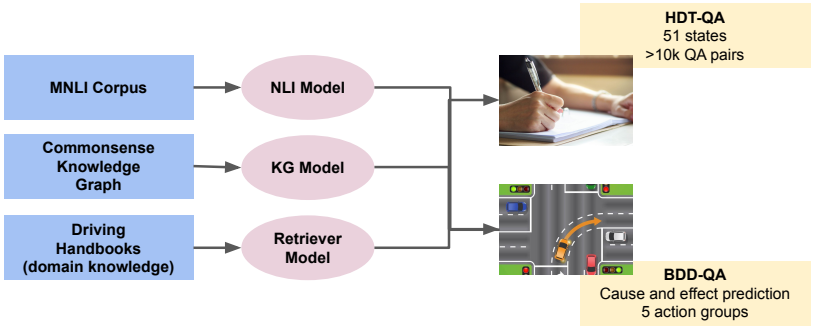

Figure 1: Overview of our study framework, which evaluates three knowledge-enhanced language methods adapted with different knowledge sources on two evaluation datasets in a zero-shot manner.

tracting commonsense and domain knowledge from structured and unstructured data sources. An overview of our study framework is shown in Figure 1.

We summarize our contributions as follows:

1. We formulate two novel text-based tasks and benchmarks for situational reasoning in the traffic domain: BDD-QA, which evaluates whether systems can apply causal reasoning to make decisions in arbitrary traffic situations, and HDT-QA, which tests the ability of systems to answer questions that require domain knowledge from driving manuals. We divide the datasets into meaningful partitions to allow balanced and fine-grained analysis.
2. We adapt different knowledge-aware methods that have been shown to generalize well across natural language reasoning tasks in prior work: natural language inference models, knowledge graph-based commonsense reasoners, and a novel retrieval-based QA method with an open-book access to official driving manuals.
3. We perform extensive experiments of the three methods with different language models as backbones against the two datasets in a zero-shot evaluation setup. We perform an in-depth analysis of model performance on data partitions and look closer into the model predictions to provide useful insights into the role of background knowledge for building robust traffic understanding methods.

Related work

Traffic Understanding

CADP (Shah et al. 2018) is a spatio-temporally annotated dataset for accident forecasting using traffic camera views. The Berkeley Deep-Drive dataset (BDD) is a video dataset consisting of real driving videos containing abundant driving scenarios (Xu et al. 2017). The follow-up work, BDD-X (Kim et al. 2018), provides the action description of the BDD video and their explanations. TrafficQA (Xu, Huang, and Liu 2021) consists of over 60K QA sets based on over 10k traffic scenes. The QA set of TrafficQA includes 6 different aspects of reasoning problems: basic understanding, attribution, introspection, counterfactual inference, event forecasting, and reverse reasoning. All QA pairs are based on visual inputs.

The traffic benchmarks have inspired corresponding methods. By using knowledge to enhance traffic understanding, Wickramarachchi, Henson, and Sheth (2020) provides a demonstration of the process of creating and evaluating knowledge graph embedding for autonomous driving data. Chowdhury et al. (2021) enhance the knowledge graph for autonomous driving and the task of scene entity prediction. ITSKG (Muppalla et al. 2017) is a knowledge graph framework for extracting actionable information from raw sensor data in traffic. CoSI (Halilaj et al. 2021) proposes a knowledge graph-based approach for representing information sources relevant to traffic situations.

Different from traffic monitoring, which mainly requires the perception modality, in this work, we explore a shift in the paradigm from monitoring to understanding, which requires abundant domain knowledge and commonsense reasoning methods. However, perceptual annotations provide valuable information that can be reused for creating comprehension tasks. We utilize the textual descriptions of actions and their explanations associated with the BDD-X dataset to construct our BDD-QA dataset for causal reasoning.

Situational Reasoning in Natural Language

MultiNLI (Williams, Nangia, and Bowman 2017) is a large (433k) NLI corpus with ten distinct genres of written and spoken English. MNLI has become a popular benchmark for most of the language models, such as BERT (Devlin et al. 2018), RoBERTa (Liu et al. 2019), BART (Lewis et al. 2019), and DeBERTa (He et al. 2020). These methods have been widely used, but have not been applied to the traffic domain, to the best of our knowledge. In our work, we use such models that are pretrained with the MNLI dataset, to evaluate their zero-shot performance on inference tasks in the traffic domain based on the assumption that the models are capable of causal reasoning generally.

ATOMIC (Sap et al. 2019) provides a large atlas of everyday commonsense reasoning with a focus on inferential if-then knowledge. CauseNet (Heindorf et al. 2020) provides a large-scale knowledge base of claimed causal relations between concepts which can be used for casual reasoning and question-answering tasks. Ma et al. (2021a) developed a method that utilizes commonsense knowledge that is extracted from the knowledge graph to train language models

Table 1: Examples of BDD-QA and HDT-QA. (*) denotes the correct answer. BDD-QA-EP questions ask for the correct effect given the cause and BDD-QA-CP for the correct cause given the effect. For HDT-QA, we show one example for each category (having 2,3,4,5 candidate answers)

BDD-QA-EP
C: The light has turned green and traffic is flowing smoothly.
E1: The driver jerks abruptly to the right;
E2: The car is driving straight into the parking lot;
E3: The car accelerates slowly to a maintained speed(*)
BDD-QA-CP
E: The car is driving forward slowly.
C1: The light ahead was red at the intersection;
C2: The passenger is seated in the car and the lane is clear;
C3: There is slow traffic in front of the car(*)
HDT-QA (2 candidates)
Q: Yield also means stop if you cannot merge safely into the flow of traffic
A1: True(*); A2: False
HDT-QA (3 candidates)
Q: You just sold your vehicle. You must notify the DMV within ___ days.
A1: 15; A2: 5(*); A3: 10;
HDT-QA (4 candidates)
Q: You have the right of way when you are:
A1: Entering a traffic circle;
A2: Backing out of a driveway;
A3: Leaving a parking space;
A4: Already in a traffic circle(*)
HDT-QA (5 candidates)
Q: Why should you cover your cargo during its transportation?
A1: To protect the cargo from weather only;
A2: There is no reason;
A3: To protect people from the cargo's contents if it spills;
A4: To protect people from the cargo's contents if it spills and to protect the cargo from the weather(*);
A5: To help your vehicle accelerate smoothly

via self-supervision and such models are proven to have high accuracy across several zero-shot tasks (Zhang et al. 2022). Given that the knowledge-driven models perform well on many general-domain zero-shot commonsense tasks, in this work, we evaluate their performance on reasoning tasks in the traffic domain to investigate the effect of commonsense knowledge on answering traffic-domain questions.

DPR (Dense Passage Retrieval) (Karpukhin et al. 2020) is a passage retrieval method using dense representations of sentences and passages for QA. Trained on a Wikipedia dump and several QA datasets, the DPR model overperforms several traditional context retrieval methods like TF-IDF and BM25. Unified-QA (Khashabi et al. 2020) is a generalized QA system across 4 formats of QA and is proven to perform well on all of the benchmarks. In our work, we retrieve relevant domain knowledge with DPR to help answer domain-specific questions with the Unified-QA model.

Tasks & Datasets

We refer to situational reasoning as the task of providing the best possible justification or decision for a given situation. For instance, in the traffic domain, the justification for "Why did the car accelerate?" could be the light turning green or the absence of traffic. We focus on situational reasoning in the traffic domain. Formally, we pose traffic situational reasoning as a multiple-choice question-answering task. Each task entry consists of a natural language question Q, and n candidate answers $\{A_1, ..., A_n\}$, consisting of 1 correct answer and $n-1$ distractors. The task of an AI model is to select the most probable answer among the candidates.

To evaluate whether AI models comprehensively understand the traffic domain knowledge, we introduce two diverse tasks: BDD-QA and HDT-QA. BDD-QA tests model understanding of the causality of traffic participant behaviors and their reasons, while HDT-QA focuses on driving policies, and legible or recommended drivers' decisions. Examples of the two datasets are shown in Table 1. Next, we describe the details of these two QA datasets.

BDD-QA dataset

Data Source We create a multiple choice question set from the Berkeley Deep-Drive Explanation Dataset (Kim et al. 2018) (BDD-X), which includes over 26k QA sets and covers an extensive range of weather and road types. Given that the dataset consists of an action and its justification (reason), we treat the action as an effect and the justification as a cause.

Sampling Strategy We use sentence transformers (Reimers and Gurevych 2019) to represent the sentences as sentence embeddings. The model we use, *all-mpnet-base-v2*, is based on MPNet (Song et al. 2020) and is pretrained on over 1B sentence pairs and reportedly reached a state-of-art performance. All of the sentence embeddings are normalized and we compute their distance using cosine similarity. To make the sentence embedding more accurate, we transform each sentence into a declarative and grammatically complete one. For example, if a justification sentence is *to turn left*, it will be transformed to *The car wants to turn left* by filling in the subject. If the justification sentence is *because the light turns red*, we will remove the *because*, resulting in the sentence *The light turns red*. To avoid having causes that are too similar, we remove the QA pairs that have a cause that has a similarity to others higher than a threshold, in our work, we set this threshold to *0.9*.

To generate multiple-choice questions, for each cause-and-effect pair, we employ two different approaches: (1) Sampling QA-pairs by dissimilar effects for effects-prediction, **(BDD-QA-EP)**, and (2) Sampling QA-pairs by dissimilar causes for causes-prediction, **(BDD-QA-CP)**. We randomly sample cause/effect from other pairs, then we set an upper bound of sentence similarity to avoid sampling too similar answers that are also correct for a question. The upper bound threshold t is set to 0.4 in our work, all the randomly sampled cause/effect whose similarity with the true answer is higher than the threshold would be removed and we will re-sample until it satisfies the threshold restriction. For example, given the cause *The road doesn't have much traffic* and effect *The car accelerates down the road*, a good effect distractor would be *The vehicle is stopped* rather than *The car is accelerating faster*. To improve the diversity of

Table 2: Examples of 5 classes of effects (car actions) discovered in the BDD-QA dataset.

Classes	Sentences
Accelerate	The car accelerates The car inches forward The car is traveling down the road.
Slow	The car slows slightly The car moves forward slowly The car maintains a slow speed
Stop	The car stops The car is parked at the right curb The car is stationary
Merge	The car merges into the lane to its left The car is moving to the left lane The car is switching lanes to the left
Turn	The car slows down and pulls to the right The car turns left and drives forward The car swerves to the left and slows

the sampled distractors, we also make sure that the dissimilarity between the distractors is also lower than the same threshold t.

Action Classification The project (Ope 2020) presents an ontology that describes the concepts in the traffic domain, including car maneuvers. In the real world, the possible actions performed by cars are few and lead to a limited set of effects. Inspired by this, we classify the effects of BDD-QA into several classes, each of which presents a type of car action. After we compute the sentence embeddings for each action, we use k-means (MacQueen 1967) for the action sentence clustering. After trying different numbers of classes, we noticed that the clusters are intuitive. We decide to use 5 classes that can represent the car's actions best from a human perspective, namely: *accelerate*, *slow*, *stop*, *merge* and *turn*. Table 2 shows examples for each of the five classes.

HDT-QA dataset

We also develop a complementary QA set by using driving tests intended to test the driving knowledge of humans. We call this dataset HDT-QA (Human Driving Test QA).

Data Source We scrap the questions of HDT-QA from a website that contains the driving test data and driving manuals from all 51 states of the USA (Dri 2022). For each state, there are three subjects of driving tests and manuals *Motorcycle*, *Car* and *Commercial Driver*, resulting in 153 state-subject combinations. The test data are all multiple-choice questions, where the number of answer candidates ranges between 2 and 5. The statistics of our HDT-QA set for each number of candidate answers are shown in Table 3.

Filtering To make our test data fair for text-based methods, we only keep questions that can fully be answered based on the textual content. We exclude all questions with images (e.g., traffic signs). We have also filtered out the questions where one of the candidate answers is *all of the above*, as answering such questions is not the focus of our work.

Table 3: Statistics of the partitions of our two datasets.

Dataset	Partition	Size
BDD-QA	Event Prediction	3,139
	Cause Prediction	3,139
HDT-QA	2 answers	131
	3 answers	2,398
	4 answers	7,558
	5 answers	40

Method

A key strength of neural language models is their ability to provide answers for open-world reasoning tasks (Devlin et al. 2018; Radford et al. 2019). While the models can be tuned on a dataset to enhance their accuracy, prior work has shown that this leads to over-fitting (Ma et al. 2021b). Anticipating similar over-fitting to QA sets in the traffic domain, we focus on developing models that include commonsense and domain knowledge to enable them to better understand situations in traffic and generalize across unseen traffic QA tasks. We propose three methods to enrich language models with background knowledge for the traffic domain: *NLI-based models*, *Knowledge-based models*, and *Retrieval-based models*. We use readily available models and run them directly on our two benchmarks in a zero-shot setting.

Inference-based reasoning models

We propose an NLI method based on semantic-level reasoning for the zero-shot evaluation of our tasks. Given a premise and its hypothesis that can be from any domain, the NLI model we use is expected to capture coherence information between them. It takes two sentences as the input and outputs a 3-digit vector which represents confidence letting the sentence pair's relation be entailment, neutral, and contradiction.

To avoid the effect of the data format, we keep our QA data as declarative sentence pairs. As is described in the last paragraph, given a sentence pair S_1, S_2, the model is asked to give the score of entailment, neutral, and contradiction of these two sentences as P_e, P_n, P_c, respectively. Then the model chooses the candidate with the highest margin score of $P_e - P_c$ as the true answer. Formally, the final prediction of the model will be:

$$pred = \arg\max_i (P(e|p_i, MNLI) - P(c|p_i, MNLI))$$

Where p_i is the i_{th} candidate sentence pair for the model to choose. e stands for entailment and c stands for contradiction. $MNLI$ refers to the inference knowledge that is extracted from the MNLI dataset by the language model.

KG-based reasoning models

Self-supervision with structured knowledge has proven to improve the language model's ability to solve reasoning problems in a zero-shot setting (Ma et al. 2021a; Zhang et al. 2022; Dou and Peng 2022). Thus, we evaluate the performance of language models enhanced by commonsense knowledge of traffic domain question-answering problems.

Our knowledge-based QA framework follows the setup of Ma et al. (2021b); Zhang et al. (2022). Given a question and several candidates, the model chooses the question-candidate pair that best entails the knowledge extracted from the knowledge graph:

$$pred = \arg\max_i(R(s_i|CSKG))$$

where $R(s|CSKG)$ is the reasonable score of a sentence based on the knowledge that the language model extracted from the Commonsense Knowledge graph.

We chose models that are pretrained on a synthetic dataset that is generated from the Commonsense Knowledge Graph (Ilievski, Szekely, and Zhang 2021). In total, Commonsense Knowledge Graph contains 7 million commonsense statements, which is a significant portion of implicit facts that the commonsense models have absorbed via self-supervision.

Retrieval-based QA models

Retrieval-based QA models aim to find evidence for the correct answer for a given contextualized question in background knowledge documents. This task aims to capture domain knowledge such as rules and regulations we abide by while driving in traffic. We pose the input to the retrieval-based models as a multiple-choice question answering based on context. The main idea is to extract the most relevant content as evidence that can help answer the question based on domain documents and use it as a context to find the best possible choice.

We propose a novel pipeline for domain-based question answering. We use Dense Passage Retriever (DPR) (Karpukhin et al. 2020) as the relevant domain extractor module to encode each paragraph and question sentence into vectors. Given a corpus C include several passages PA, we use a cosine similarity score to extract the most relevant paragraph:

$$PA_{select} = \arg\max_{PA_i} \ cos\_sim(PA_i, Q), PA_i \in C$$

The passage extracted then along with the question and choices are passed to a QA model, then the QA model generates the predicted answer, the probability of generating an answer is:

$$P(GA|Q, A, C) = \prod_{i=i}^{n} P(ga_i|Q, A, PA_{select}, ga_1, ga_2, ..., ga_{i-1})$$

Where C is the whole corpus, for our task, we use a comprehensive collection of driving manuals as the domain corpus (Dri 2022). After processing the manual PDFs, we divide them into sentences by simply using the punctuation *" . ? ! "* as sentence boundaries. Then we use a T5-based grammar correction (Gra 2021) as a data cleaning tool to make our corpus have correct spelling and grammar. Finally, we combine every 10 neighborhood sentences as paragraphs that are fed to the QA model.[2] In total, there are 42.979 paragraphs and each paragraph has an average word count of 147.5.

After the model generates the predicted answer, we choose the answer which is most similar to the GA as our final prediction.

[2]Paragragh segmentation based on the similarity between neighboring sentences yielded consistently worse results.

Experiments

Model setup

For Inference-base reasoning models, we select several NLI models that are trained on the MNLI (Williams, Nangia, and Bowman 2017) dataset. These models are: *Roberta-large*, *Bart-large*, *Deberta-large*, *Deberta-v2-xlarge* and *Deberta-v2-xxlarge*.

For KG-based reasoning models, we choose the five models from Zhang et al. (2022) for evaluation: Roberta-base, Roberta-large and T5-small, T5-large, and T5-3b models.

For Retrieval-based models, we use the sentence transformer model pretrained in (Karpukhin et al. 2020) to encode the paragraphs and sentences. As a QA model, we use the Unified-QA T5-based transformer model (Khashabi, Kordi, and Hajishirzi 2022), to output the most probable candidate answer.

Other Baselines

To indicate the upper bound of models' performance on our dataset, we have also tested a supervised method based on tuning Roberta-large. We have divided all of the datasets into training and testing parts with a ratio of 90% and 10%. For the HDT-QA dataset with 2 and 5 questions, we do not perform the supervised method due to the limited size of the dataset.

To investigate the correlation between our two datasets, we also include a transfer learning experiment that takes the supervised model trained on one dataset to evaluate the other dataset. We also report the results of an unsupervised vanilla Roberta-large model and vanilla Unified-QA model as well as of a random baseline.

Results

We have tested three methods on our two datasets. We use accuracy as our evaluation metric. For the BDD-QA dataset, the Unified-QA model reaches the best performance by adapting multiple formats of question answers. For the HDT-QA dataset, our proposed retrieval-based model, DPR+Unified-QA outperforms other methods by nearly 20 points, which shows the effectiveness of retrieving domain knowledge for question answering.

BDD-QA Task

Table 4 shows the results for three different types of models (NLI-based, KG-based, and Retrieval-based QA) on the BDD-QA dataset. The result shows that the best accuracy is obtained by Unified-QA-v2, which benefits from its adaption of multiple formats of question answers. Such performance is 29.6 points higher than the unsupervised Roberta-large result, but still, has a large gap with the supervised model (16.5 points). In general, NLI-based models have a

Table 4: Evaluation results on the BDD-QA-EP and BDD-QA-CP dataset of our three methods. The Vanilla Roberta-large model was only pretrained by unlabelled corpus. The Vanilla Unified-QA method doesn't feed the most relevant corpus to the model. Different from other results, supervised models only use 10% of the whole data for evaluation. The result of the human evaluation is included in the last row.

Method	**Model**	BDD-QA-EP	BDD-QA-CP	Avg
Random	-	33.3	33.3	33.3
Vanilla models	Roberta-large	36.1	43.6	40.0
	UnifiedQA-v2	**75.9**	63.3	**69.6**
NLI-based	Roberta-large-mnli	63.2	58.6	60.9
	Bart-large-mnli	66.3	56.8	61.5
	Deberta-large-mnli	67.0	54.5	60.8
	Deberta-v2-xlarge-mnli	71.3	59.7	65.5
	Deberta-v2-xxlarge-mnli	72.4	**64.9**	68.7
Knowledge-based	Roberta-base	63.3	54.4	58.9
	Roberta-large	71.7	56.8	64.3
	T5-small	54.3	46.0	50.1
	T5-large	66.4	55.5	61.0
	T5-3b	71.0	58.6	64.8
Retrieval-based	UnifiedQA-v2 + DPR	71.4	55.7	63.6
Supervised	Roberta-large	93.0	81.2	86.1
Human	-	74.0	68.0	71.0

Table 5: Evaluation results on the BDD-QA-EP dataset of our models on 5 action classes.

Methods		**Action Class**				
		Accelerate	Slow	Stop	Merge	Turn
NLI-based	Roberta-large	70.5	59.7	68.5	59.6	51.8
	Bart-large	77.1	59.5	67.5	68.6	57.1
	Deberta-large	74.7	58.4	71.6	70.1	58.7
	Deberta-v2-xlarge	74.5	62.5	**76.2**	74.5	66.2
	Deberta-v2-xxlarge	**79.0**	65.3	75.8	72.2	65.7
Knowledge-based	Roberta-base	56.5	57.4	51.5	66.5	**82.4**
	Roberta-large	68.7	**70.4**	68.5	**77.0**	81.3
	T5-small	57.1	58.8	59.0	42.6	58.0
	T5-large	59.7	60.8	65.8	65.5	80.1
	T5-3B	65.1	68.7	63.7	75.1	81.8

balanced performance in the two datasets while KG-based models perform well on the BDD-QA-EP dataset but poorly on BDD-QA-CP.

Zero-shot performance depends on model size and background training data. We see that the performance of NLI-based models is getting continuously better when the model size grows, which is expected. Specifically, NLI-based models' performance on BDD-QA-EP is growing faster when the model size is smaller, while for BDD-QA-CP the growth is more obvious between bigger models. Given the sentence embeddings of all the causes and effects, we have computed the average sentence similarity between causes as 0.403, while for effects it is 0.553, which means causes are more different from each other than the effects. The difference of similarities is intuitive because the effects (cars' actions) are limited while the causes represent factors in the environment and are therefore more open-ended. By focusing on three Deberta-large models from Table 4, we observe that the performance on BDD-QA-CP grows faster than BDD-QA-EP when the model size is bigger, $(64.9 - 59.7 > 72.4 - 71.3$ while $59.7 - 54.5 \approx 71.3 - 67.0)$. Such a result implies that predicting the correct cause given an effect needs a language model having a larger capacity, which is coherent with the observation between cause and effect prediction.

Among all of the knowledge-based models, T5-3b, the best-performing model across five zero-shot benchmarks in Zhang et al. (2022), which is an encoder-decoder model trained with sequence-to-sequence language model loss, achieves the best performance. Roberta-large, the encoder-only model trained on masked language model loss, also reaches a high performance that is close to T5-3b's. Overall, knowledge-based models are not good at predicting the cause, none of them reaches 60% accuracy in the BDD-QA-CP dataset. To explain this low performance, we look deeper into the properties of the synthetic training data that was constructed in (Ma et al. 2021a) from CSKG (Ilievski, Szekely, and Zhang 2021). The source graph has abundant information about X causes Y, which we translate into training questions of the form: *What does X cause?* then the different effects are sampled. However, there are no questions about the inverse formulation *What is the cause of X?* like what we do to the BDD-QA-CP dataset. Such a result is consistent with the finding in Zhang et al. (2022) that the language models learn to answer well about what is in the data

and do not generalize well to other aspects. Finally, there is still a large gap between both models' accuracy and that of the supervised Roberta model, which leaves plenty of space for future improvement of the models.

Regarding the retriever-based models, Unified QA itself benefits from its adaption of multiple formats of question-answer pairs so its performance is close to the best one among the NLI-based and KG-based models. However, after we provide the most relevant passage with it, the performance decreases. This implies that knowledge in the driving handbook provides little help for the model to answer the BDD-QA questions.

Overlap analysis. We expect that the language model performance highly depends on their training data and training methods. To investigate the predictions of two different kinds of models, we look closer at the predictions of the best-performing NLI-based model and KG-based model Deberta-v2-xxlarge and Roberta-large, respectively. Among 3,139 QA pairs in BDD-QA-EP, Deberta-v2-xxlarge can correctly answer 2,256 questions and Roberta can predict 2,302 QA pairs correctly. However, there are only 1,826 answers that are correctly answered jointly, which means that for an ensemble model that can combine the two models' correct answers, the accuracy would be 87.0% on BDD-QA-EP. For BDD-QA-CP, the combined accuracy would be 79.1%. Such results are much higher than the performance of the two models alone and are close to the result of supervised learning. Such results encourage us to investigate the granular performance of two kinds of models by splitting the dataset into several classes. We discuss this next.

Result on classes of effects. We investigate the granular performance of different models on the action (cause) classes of the BDD-QA-EP dataset. Table 5 shows that the Knowledge-based models perform best on the *Turn*, *Merge* and *Slow* classes, while NLI-based models do better on the other two classes (*Move* and *Stop*). In the real world, turning, merging, and slowing are more complicated actions than moving and stopping because their causes vary a lot. For example, if a car merges to the left, the cause might be *The snow is taking up too much of the right lane*, or *There are pedestrians in the right* or *The car is preparing to execute a u-turn maneuver*, or a long list of other causes. But for Accelerate, the list of causes is more narrow: it would either be *The light turned green* or *The road is clear*. We conclude that knowledge-based models can handle complex reasoning tasks in the traffic domain, while NLI-based models are better at surface-level inference. Here, again we see that the performance for most action classes grows with the increase in model size.

Human evaluation. Since BDD-QA is constructed by an automatic procedure, we perform a human evaluation for this dataset. For each partition of the dataset, we randomly select 50 QA pairs. Besides providing an answer for each question, we also ask the annotators to give confidence in their results on a 5-point Likert scale (1 being low and 5 being high) to check whether the questions are commonsensical. Each question is answered by 3 humans and we choose the majority-voted candidate as the correct answer. The result is shown in the last row of Table 4. We note that the human score is 1.4% points higher than the best zero-shot result, and the average annotator confidence is 3.67. We are surprised by the relatively low human score on this task. Upon further analysis, we notice that most of the questions that are wrongly answered by humans have multiple reasonable answers, e.g., given the cause *The cross traffic has cleared and the pedestrian is gone* The correct effect is *The car makes the left turn and heads down the street*, but another effect *The car very slow accelerates* is also reasonable given the cause. For this question, all three humans chose the latter answer. In the future, we will revise the dataset to further clean up the noise.

HDT-QA Task

As shown in Table 6, all of the NLI-based and KG-based models perform consistently poorly on the HDT-QA dataset. The growth of the performance together with the model size that was observed on the BDD-QA dataset is not as clearly visible on the HDT-QA dataset. Such a result implies that obtaining basic commonsense knowledge and semantic inference information is not enough for answering human driving test questions. The examples of HDT-QA, shown in Table 2, are about the details of the policies of driving. It is hard even for humans to pass the driving exams without learning specific rules and driving policies captured in domain manuals. As a result, it is intuitive to retrieve domain knowledge to help language models better answer the human driving test.

As shown in Table 6, UnifiedQA itself performs slightly better than the NLI-based and KG-based models. After we provide the most relevant paragraph to the QA model automatically with the DPR retriever, the average performance of the subsets improves by 14.2% points. This means the QA system is able to capture relevant domain knowledge efficiently for answering human driving test questions, especially when combined with a dense retrieval model that can provide the most relevant set of sentences that serve as evidence.

The performance of unsupervised learning and transfer is close to that of NLI-based and KG-based models, which means the NLI, KG, and BDD-enhanced models contribute little for models to answer the HDT-QA questions. Lastly, supervised learning reaches a high accuracy, which is intuitive because some of the questions in the training and the test set share the same knowledge.

Discussion

Our experiments show that language models trained with semantic level coherence information and commonsense knowledge extracted from knowledge graphs are both performing well at answering traffic domain reasoning tasks. Careful domain information extraction and model adaption leads to significant improvement of language models' performance on domain-specific QA sets. Next, we discuss and list future extensions inspired by our results.

More efficient retrieving of domain knowledge In this paper, we use Dense Passage Retrieval to capture the most relevant paragraph in the driving manuals and directly fed

Table 6: Evaluation results on the HDT-QA dataset of 10 models. We only train the HDT-QA-3Q and HDT-QA-4Q for supervised learning because the size of HDT-QA-2Q and HDT-QA-5Q are both limited.

Methods		HDT-QA-2Q	HDT-QA-3Q	HDT-QA-4Q	HDT-QA-5Q	Avg
Random	-	50.0	33.3	25.0	20.0	25.7
Vanilla models	Roberta-large	55.4	43.7	36.7	9.10	36.2
	UnifiedQA-v2	67.9	51.6	44.0	25.0	47.1
NLI-based	Roberta-large-mnli	63.4	41.0	33.0	15.0	38.1
	Bart-large-mnli	65.6	46.1	34.3	22.5	42.2
	Deberta-large-mnli	62.6	42.8	34.5	15.0	38.7
	Deberta-v2-xlarge-mnli	62.6	45.7	36.3	17.5	40.5
	Deberta-v2-xxlarge-mnli	63.4	47.5	36.1	22.5	42.4
Knowledge-based	Roberta-base	60.3	36.0	29.0	25.0	37.6
	Roberta-large	63.4	38.9	30.8	27.5	40.1
	T5-small	57.3	35.4	23.8	22.5	34.7
	T5-large	64.1	40.3	24.6	22.5	37.9
	T5-3b	66.4	45.0	25.1	22.5	39.7
Retrieval-based	UnifiedQA-v2 + DPR	**80.2**	**57.3**	**47.7**	**60.0**	**61.3**
Supervised	Roberta-large	-	71.7	75.8	-	73.8
Transfer learning	Roberta-large	45.8	43.2	37.9	21.8	37.2

the paragraph to the QA system. However, the language models are not learning the policies, they are answering the questions only by consulting the driving manual simultaneously in an open-book setting, which can be misled by vague information or context mismatch. Developing methods for effectively extracting domain knowledge and letting language models learn this knowledge is an important future work for solving HDT-QA and similar traffic-specific QA tasks. Selecting informative sentences or building knowledge graphs based on raw data are possible methods for a more precise injection of domain knowledge.

Joint reasoning over domain knowledge and causal commonsense knowledge Our methods have isolated the different knowledge types and have focused on learning either causal or domain knowledge, but not both. We anticipate that a robust AI model should be able to reason over both knowledge types simultaneously and know which questions require which specific knowledge statements. A key future item is integrating the two knowledge types together in a joint reasoning system and developing benchmarks that can evaluate both aspects in a more coherent manner.

From natural language QA to multi-modal traffic QA Our study focused on understanding traffic situations based on using only text as background information. However, visual information in images and videos is a key component of traffic scenes. Automatic driving systems take images as input and make decisions according to them. Popular datasets like Traffic-QA (Xu, Huang, and Liu 2021) are also based on background information that is given by an image. Considering visual information is necessary for building a holistic robust system that understands traffic situations. Thus, we believe that multi-modal benchmarks and methods that combine perception and background knowledge for situational traffic understanding are the next milestones toward developing real-world neuro-symbolic traffic models.

Conclusion

Besides robust perception models, a holistic situational understanding of traffic requires generalizable models equipped with commonsense and domain knowledge, which has received little attention in prior work. To bridge this gap, this paper studied the strengths and weaknesses of current zero-shot adaptation methods and their granular effects on different partitions of tasks. We constructed two benchmarks for evaluating the ability of zero-shot causal reasoning (BDD-QA) and accessing domain knowledge (HDT-QA) to answer questions about traffic situations. We proposed two representative methods that focus on zero-shot causal reasoning and one novel pipeline for retrieving domain information. On the BDD-QA dataset, we have observed the limited prediction overlap between the two models. To deeper investigate that, we split the actions of the cars into several categories, observing significant differences in performance between the methods, which provides a convincing explanation of the result of the limited overlap. On the HDT-QA dataset, we observed a giant performance improvement after we fed the relevant information extracted from driving manuals in an open-book setting.

The methods in our work can be easily extended to other traffic and related tasks in other domains. Our code and data can be downloaded at https://github.com/saccharomycetes/text-based-traffic. Key future directions include more efficient knowledge retrieval, better integration of causal and domain knowledge, and the development of multi-modal traffic benchmarks and methods.

Acknowledgements

We would like to thank all of the anonymous reviewers for their valuable suggestions. We also thank Zhivar Sourati, Yifan Jiang, Jiachen Zhang, Thiloshon Nagarajah, Xiaoshu Luo, Riley Li, and Prateek Chhikara for participating in our human evaluation study.

References

2020. ASAM OpenXOntology. https://www.asam.net/project-detail/asam-openxontology/.

2021. Fine-Tune a Transformer Model for Grammar Correction. https://www.vennify.ai/fine-tune-grammar-correction/.

2022. Free DMV Permit Practice Tests. https://www.dmv-test-pro.com/.

2022. Intelligent Traffic Management System Market Worth $27.92 Billion By 2030. https://www.grandviewresearch.com/press-release/global-intelligent-traffic-management-system-market.

Chowdhury, S. N.; Wickramarachchi, R.; Gad-Elrab, M. H.; Stepanova, D.; and Henson, C. A. 2021. Towards Leveraging Commonsense Knowledge for Autonomous Driving. In *ISWC (Posters/Demos/Industry)*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dou, Z.-Y.; and Peng, N. 2022. Zero-shot Commonsense Question Answering with Cloze Translation and Consistency Optimization. *arXiv preprint arXiv:2201.00136*.

Halilaj, L.; Dindorkar, I.; Lüttin, J.; and Rothermel, S. 2021. A knowledge graph-based approach for situation comprehension in driving scenarios. In *European Semantic Web Conference*, 699–716. Springer.

He, P.; Liu, X.; Gao, J.; and Chen, W. 2020. DeBERTa: Decoding-enhanced BERT with Disentangled Attention.

Heindorf, S.; Scholten, Y.; Wachsmuth, H.; Ngonga Ngomo, A.-C.; and Potthast, M. 2020. Causenet: Towards a causality graph extracted from the web. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 3023–3030.

Ilievski, F.; Szekely, P.; and Zhang, B. 2021. Cskg: The commonsense knowledge graph. In *European Semantic Web Conference*, 680–696. Springer.

Karpukhin, V.; Oğuz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W.-t. 2020. Dense Passage Retrieval for Open-Domain Question Answering.

Khashabi, D.; Kordi, Y.; and Hajishirzi, H. 2022. UnifiedQA-v2: Stronger Generalization via Broader Cross-Format Training.

Khashabi, D.; Min, S.; Khot, T.; Sabharwal, A.; Tafjord, O.; Clark, P.; and Hajishirzi, H. 2020. UnifiedQA: Crossing Format Boundaries With a Single QA System. arXiv:2005.00700.

Kim, J.; Rohrbach, A.; Darrell, T.; Canny, J.; and Akata, Z. 2018. Textual Explanations for Self-Driving Vehicles. *Proceedings of the European Conference on Computer Vision (ECCV)*.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2019. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach.

Ma, K.; Ilievski, F.; Francis, J.; Bisk, Y.; Nyberg, E.; and Oltramari, A. 2021a. Knowledge-driven Data Construction for Zero-shot Evaluation in Commonsense Question Answering. In *AAAI*.

Ma, K.; Ilievski, F.; Francis, J.; Ozaki, S.; Nyberg, E.; and Oltramari, A. 2021b. Exploring Strategies for Generalizable Commonsense Reasoning with Pre-trained Models. *EMNLP 2021*.

MacQueen, J. 1967. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, 281–297.

Muppalla, R.; Lalithsena, S.; Banerjee, T.; and Sheth, A. 2017. A knowledge graph framework for detecting traffic events using stationary cameras. In *Proceedings of the 2017 ACM on Web Science Conference*, 431–436.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. arXiv:1908.10084.

Sap, M.; Le Bras, R.; Allaway, E.; Bhagavatula, C.; Lourie, N.; Rashkin, H.; Roof, B.; Smith, N. A.; and Choi, Y. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 3027–3035.

Shah, A. P.; Lamare, J.-B.; Nguyen-Anh, T.; and Hauptmann, A. 2018. CADP: A novel dataset for CCTV traffic camera based accident analysis. In *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–9. IEEE.

Song, K.; Tan, X.; Qin, T.; Lu, J.; and Liu, T.-Y. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33: 16857–16867.

Wickramarachchi, R.; Henson, C.; and Sheth, A. 2020. An evaluation of knowledge graph embeddings for autonomous driving data: Experience and practice. *arXiv preprint arXiv:2003.00344*.

Williams, A.; Nangia, N.; and Bowman, S. R. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.

Xu, H.; Gao, Y.; Yu, F.; and Darrell, T. 2017. End-to-end learning of driving models from large-scale video datasets. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2174–2182.

Xu, L.; Huang, H.; and Liu, J. 2021. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9878–9888.

Zhang, J.; Ilievski, F.; Ma, K.; Francis, J.; and Oltramari, A. 2022. An Empirical Investigation of Commonsense Self-Supervision with Knowledge Graphs. *arXiv preprint arXiv:2205.10661*.

Appendix

Details of chosen models

For the Inference-base reasoning model, we select models trained on the same dataset for a fair comparison, and we specifically chose MNLI because it contains 10 different genres and a very large corpus (433k). We directly download the model from hugging-face, all of them share the same input and output format (take in a premise and hypothesis, give the confidence of the sentence pair being "entail", "neutral", "contract"). In general, we test the model's ability to make the right inference on the traffic domain without any fine-tuning.

For KG-based reasoning models, the chosen models have two different model architectures and all have different model sizes. The encoder-only model, Roberta, uses the Masked Language Model loss to give a score for each question-candidate pair. After getting the score for each candidate, the Roberta models are trained with the margin loss function: $L_{margin} = \frac{1}{n}\sum^{n}_{\substack{i=1 \\ i \neq y}} max(0, \eta - S_y + S_i)$, where S_y and S_i are the negative averaged loss for the correct answer and the distractor respectively. During the evaluation, the model chooses the pair with the highest score as the correct answer. The encoder-decoder model T5 learns to give an output '1' or '2' which represents the confidence of the model in predicting whether the given sentence pair is reasonable or not. For each candidate, its score is defined as $S = L_{true} - L_{false}$, then the training and evaluating process is the same as the encoder-only models. We splice the cause and effect together as the input of the models when we test the BDD-QA-CP and BDD-QA-EP and HDT-QA datasets. For the HDT-QA dataset, if there is an underscore appearing in the question, we fill the underscore with candidates to make complete sentences.

For Retrieval-based models, we choose the sentence transformer version $facebook - dpr - ctx_encoder - single-nq-base$, $facebook-dpr-question-encoder-single-nq-base$ to encode the paragraphs and questions, respectively. For the unified-QA model, we use the version of Unified-QA-v2, T5-3B, which is pretrained on 20 QA datasets.

Implementation Details

Regarding libraries, we used python 3.7.10, PyTorch 1.9.0, and transformers 4.11.3.

Among all the supervised learning sets, we are using a learning rate of $1e^{-5}$, batch size of 32, weight decay 0.01, training epochs of 10, adam-epsilon of $1e^{-6}$, $\beta 1 = 0.9, \beta 2 = 0.98$, the warm-up proportion of 0.05, the margin of 1.0

For CPUs, we used Intel(R) Xeon(R) Gold 5217 CPU @ 3.00GHz (32 CPUs, 8 cores per socket, 263GB ram).

For GPUs, we used Nvidia Quadro RTX 8000.