The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Augmenting End-to-End Dialogue Systems with Commonsense Knowledge

Tom Young,[1] **Erik Cambria,**[2] **Iti Chaturvedi,**[2]
Hao Zhou,[3] **Subham Biswas,**[2] **Minlie Huang**[3]
[1]School of Information and Electronics, Beijing Institute of Technology, China
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[3]Department of Computer Science and Technology, Tsinghua University, China
tom@sentic.net, cambria@ntu.edu.sg, iti@ntu.edu.sg
tuxchow@gmail.com, subham@sentic.net, aihuang@tsinghua.edu.cn

Abstract

Building dialogue systems that can converse naturally with humans is a challenging yet intriguing problem of artificial intelligence. In open-domain human-computer conversation, where the conversational agent is expected to respond to human utterances in an interesting and engaging way, commonsense knowledge has to be integrated into the model effectively. In this paper, we investigate the impact of providing commonsense knowledge about the concepts covered in the dialogue. Our model represents the first attempt to integrating a large commonsense knowledge base into end-to-end conversational models. In the retrieval-based scenario, we propose a model to jointly take into account message content and related commonsense for selecting an appropriate response. Our experiments suggest that the knowledge-augmented models are superior to their knowledge-free counterparts.

Introduction

In recent years, data-driven approaches to building conversation models have been made possible by the proliferation of social media conversation data and the increase of computing power. By relying on a large number of message-response pairs, the Seq2Seq framework (Sutskever, Vinyals, and Le 2014) attempts to produce an appropriate response based solely on the message itself, without any memory module.

In human-to-human conversations, however, people respond to each other's utterances in a meaningful way not only by paying attention to the latest utterance of the conversational partner itself, but also by recalling relevant information about the concepts covered in the dialogue and integrating it into their responses. Such information may contain personal experience, recent events, commonsense knowledge and more (Figure 1). As a result, it is speculated that a conversational model with a "memory look-up" module can mimic human conversations more closely (Ghazvininejad et al. 2017; Bordes and Weston 2016). In open-domain human-computer conversation, where the model is expected to respond to human utterances in an interesting and engaging way, commonsense knowledge has to be integrated into the model effectively.

In the context of artificial intelligence (AI), commonsense knowledge is the set of background information that an individual is intended to know or assume and the ability to use it when appropriate (Minsky 1986; Cambria et al. 2009; Cambria and Hussain 2015). Due to the vastness of such kind of knowledge, we speculate that this goal is better suited by employing an external memory module containing commonsense knowledge rather than forcing the system to encode it in model parameters as in traditional methods.

In this paper, we investigate how to improve end-to-end dialogue systems by augmenting them with commonsense knowledge, integrated in the form of external memory. The remainder of this paper is as follows: next section proposes related work in the context of conversational models and commonsense knowledge; following, a section describes the proposed model in detail; later, a section illustrates experimental results; finally, the last section proposes concluding remarks and future work.

Related Work

Conversational Models

Data-driven conversational models generally fall into two categories: retrieval-based methods (Lowe et al. 2015b; 2016a; Zhou et al. 2016), which select a response from a predefined repository, and generation-based methods (Ritter, Cherry, and Dolan 2011; Serban et al. 2016; Vinyals and Le 2015), which employ an encoder-decoder framework where the message is encoded into a vector representation and, then, fed to the decoder to generate the response. The latter is more natural (as it does not require a response repository) yet suffers from generating dull or vague responses and generally needs a great amount of training data.

The use of an external memory module in natural language processing (NLP) tasks has received considerable attention recently, such as in question answering (Weston et

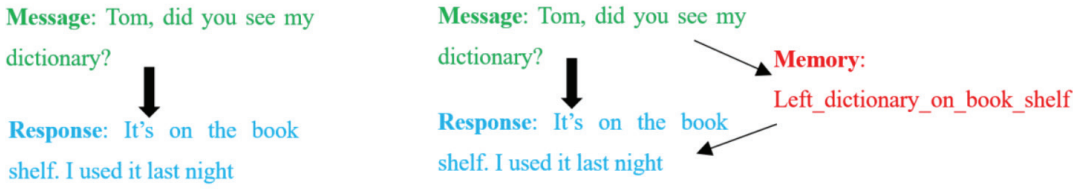

Figure 1: Left: In traditional dialogue systems, the response is determined solely by the message itself (arrows denote dependencies). Right: The responder recalls relevant information from memory; memory and message content jointly determine the response. In the illustrated example, the responder retrieves the event "Left_dictionary_on_book_shelf" from memory, which triggers a meaningful response.

al. 2015) and language modeling (Sukhbaatar et al. 2015). It has also been employed in dialogue modeling in several limited settings. With memory networks, (Dodge et al. 2015) used a set of fact triples about movies as long-term memory when modeling reddit dialogues, movie recommendation and factoid question answering. Similarly in a restaurant reservation setting, (Bordes and Weston 2016) provided local restaurant information to the conversational model.

Researchers have also proposed several methods to incorporate knowledge as external memory into the Seq2Seq framework. (Xing et al. 2016) incorporated the topic words of the message obtained from a pre-trained latent Dirichlet allocation (LDA) model into the context vector through a joint attention mechanism. (Ghazvininejad et al. 2017) mined FoodSquare tips to be searched by an input message in the food domain and encoded such tips into the context vector through one-turn hop. The model we propose in this work shares similarities with (Lowe et al. 2015a), which encoded unstructured textual knowledge with a recurrent neural network (RNN). Our work distinguishes itself from previous research in that we consider a large heterogeneous commonsense knowledge base in an open-domain retrieval-based dialogue setting.

Commonsense Knowledge

Several commonsense knowledge bases have been constructed during the past decade, such as ConceptNet (Speer and Havasi 2012) and SenticNet (Cambria et al. 2016). The aim of commonsense knowledge representation and reasoning is to give a foundation of real-world knowledge to a variety of AI applications, e.g., sentiment analysis (Poria et al. 2015), handwriting recognition (Wang et al. 2013), e-health (Cambria et al. 2010), aspect extraction (Poria et al. 2016), and many more. Typically, a commonsense knowledge base can be seen as a *semantic network* where *concepts* are nodes in the graph and *relations* are edges (Figure 2). Each $<concept1, relation, concept2>$ triple is termed an *assertion*.

Based on the Open Mind Common Sense project (Singh et al. 2002), ConceptNet not only contains objective facts such as "Paris is the capital of France" that are constantly true, but also captures informal relations between common concepts that are part of everyday knowledge such as "A dog is a pet". This feature of ConceptNet is desirable in our experiments, because the ability to recognize the informal relations

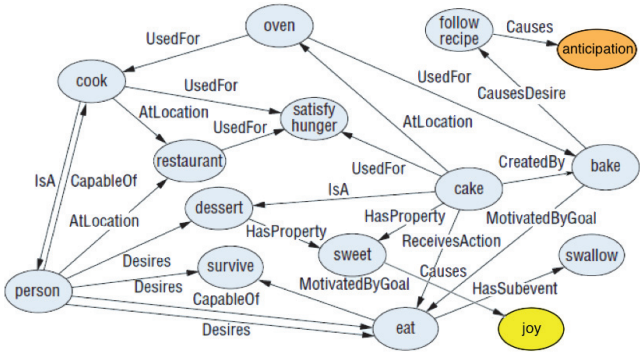

Figure 2: A sketch of SenticNet semantic network.

between common concepts is necessary in the open-domain conversation setting we are considering in this paper.

Model Description

Task Definition

In this work, we concentrate on integrating commonsense knowledge into retrieval-based conversational models, because they are easier to evaluate (Liu et al. 2016; Lowe et al. 2016a) and generally take a lot less data to train. We leave the generation-based scenario to future work.

Message (*context*) x and *response* y are a sequence of tokens from vocabulary V. Given x and a set of response candidates $[y_1, y_2, y_3..., y_K] \in Y$, the model chooses the most appropriate response $\hat{y}$ according to:

$$\hat{y} = \arg\max_{y \in Y} f(x, y), \tag{1}$$

where $f(x, y)$ is a scoring function measuring the "compatibility" of x and y. The model is trained on $<message, response, label>$ triples with cross entropy loss, where $label$ is binary indicating whether the $<message, response>$ pair comes from real data or is randomly combined.

Dual-LSTM Encoder

As a variation of vanilla RNN, a long short-term memory (LSTM) network (Hochreiter and Schmidhuber 1997) is good at handling long-term dependencies and can be used

to map an utterance to its last hidden state as fixed-size embedding representation. The Dual-LSTM encoder (Lowe et al. 2015b) represents the message x and response y as fixed-size embeddings $\vec{x}$ and $\vec{y}$ with the last hidden states of the same LSTM. The compatibility function of the two is thus defined by:

$$f(x, y) = \sigma(\vec{x}^T W \vec{y}), \tag{2}$$

where matrix $W \in \mathcal{R}^{D \times D}$ is learned during training.

Commonsense Knowledge Retrieval

In this paper, we assume that a commonsense knowledge base is composed of assertions A about concepts C. Each assertion $a \in A$ takes the form of a triple $<c_1, r, c_2>$, where $r \in R$ is a *relation* between c_1 and c_2, such as *IsA*, *CapableOf*, etc. c_1, c_2 are concepts in C. The relation set R is typically much smaller than C. c can either be a single word (e.g., "dog" and "book") or a multi-word expression (e.g., "take_a_stand" and "go_shopping"). We build a dictionary H out of A where every concept c is a key and a list of all assertions in A concerning c, i.e., $c = c_1$ or $c = c_2$, is the value. Our goal is to retrieve commonsense knowledge about every concept covered in the message.

We define A_x as the set of commonsense assertions concerned with message x. To recover concepts in message x, we use simple n-gram matching ($n \leq N$)[1]. Every n-gram in c is considered a potential concept[2]. If the n-gram is a key in H, the corresponding value, i.e., all assertions in A concerning the concept, is added to A_x (Figure 4).

Tri-LSTM Encoder

Our main approach to integrating commonsense knowledge into the conversational model involves using another LSTM for encoding all assertions a in A_x, as illustrated in Figure 3. Each a, originally in the form of $<c_1, r, c_2>$, is transformed into a sequence of tokens by chunking c_1, c_2, concepts which are potentially multi-word phrases, into $[c_{11}, c_{12}, c_{13}...]$ and $[c_{21}, c_{22}, c_{23}...]$. Thus, $a = [c_{11}, c_{12}, c_{13}..., r, c_{21}, c_{22}, c_{23}...]$.

We add R to vocabulary V, that is, each r in R will be treated like any regular word in V during encoding. We decide not to use each concept c as a unit for encoding a because C is typically too large (>1M). a is encoded as embedding representation $\vec{a}$ using another LSTM. Note that this encoding scheme is suitable for any natural utterances containing commonsense knowledge[3] in addition to well-structured assertions. We define the *match score* of assertion a and response y as:

$$m(a, y) = \vec{a}^T W_a \vec{y}, \tag{3}$$

where $W_a \in \mathcal{R}^{D \times D}$ is learned during training. Commonsense assertions A_x associated with a message is usually large (>100 in our experiment). We observe that in a lot of cases of open-domain conversation, response y can be seen as triggered by certain perception of message x defined by one or more assertions in A_x, as illustrated in Figure 4. We can see the difference between message and response pair when commonsense knowledge is used. For example, the word 'Insomnia' in the message is mapped to the commonsense assertion 'Insomnia, IsA, sleep_problem'. The appropriate response is then matched to 'sleep_problem' that is 'go to bed'. Similarly, the word 'Hawaii' in the message is mapped to the commonsense assertion 'Hawaii, UsedFor, tourism'. The appropriate response is then matched to 'tourism' that is 'enjoy vacation'. In this way, new words can be mapped to the commonly used vocabulary and improve response accuracy.

Our assumption is that A_x is helpful in selecting an appropriate response y. However, usually very few assertions in A_x are related to a particular response y in the open-domain setting. As a result, we define the *match score* of A_x and y as

$$m(A_x, y) = \max_{a \in A_x} m(a, y), \tag{4}$$

that is, we only consider the commonsense assertion a with the highest match score with y, as most of A_x are not relevant to y. Incorporating $m(A_x, y)$ into the Dual-LSTM encoder, our Tri-LSTM encoder model is thus defined as:

$$f(x, y) = \sigma(\vec{x}^T W \vec{y} + m(A_x, y)), \tag{5}$$

i.e., we use simple addition to supplement x with A_x, without introducing a mechanism for any further interaction between x and A_x. This simple approach is suitable for response selection and proves effective in practice.

The intuition we are trying to capture here is that an appropriate response y should not only be compatible with x, but also related to certain memory recall triggered by x as captured by $m(A_x, y)$. In our case, the memory is commonsense knowledge about the world. In cases where $A_x = \emptyset$, i.e., no commonsense knowledge is recalled, $m(A_x, y) = 0$ and the model degenerates to Dual-LSTM encoder.

Comparison Approaches

Supervised Word Embeddings We follow (Bordes and Weston 2016; Dodge et al. 2015) and use supervised word embeddings as a baseline. Word embeddings are most well-known in the context of unsupervised training on raw text as in (Mikolov et al. 2013), yet they can also be used to score message-response pairs. The embedding vectors are trained directly for this goal. In this setting, the "compatibility" function of x and y is defined as:

$$f(x, y) = \vec{x}^T \vec{y} \tag{6}$$

In this setting, $\vec{x}, \vec{y}$ are bag-of-words embeddings. With retrieved commonsense assertions A_x, we embed each $a \in A_x$ to bag-of-words representation $\vec{a}$ and have:

$$f(x, y) = \vec{x}^T \vec{y} + \max_{a \in A_x} \vec{a}^T \vec{y}. \tag{7}$$

This linear model differs from Tri-LSTM encoder in that it represents an utterance with its bag-of-words embedding instead of RNNs.

[1] More sophisticated methods such as *concept parser* (Rajagopal et al. 2013) are also possible. Here, we chose n-gram for better speed and recall. N is set to 5.

[2] For unigrams, we exclude a set of stopwords. Both the original version and stemmed version of every word are considered.

[3] Termed *surface text* in ConceptNet.

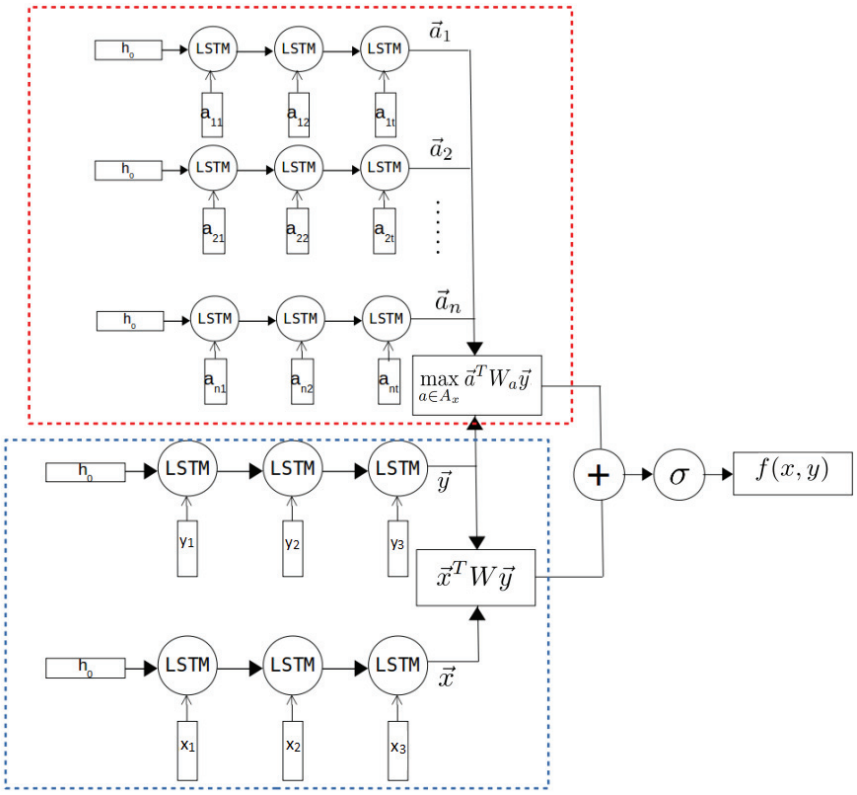

Figure 3: Tri-LSTM encoder. We use LSTM to encode message, response and commonsense assertions. LSTM weights for message and response are tied. The lower box is equal to a Dual-LSTM encoder. The upper box is the memory module encoding all commonsense assertions.

Memory Networks Memory networks (Sukhbaatar et al. 2015; Weston, Chopra, and Bordes 2014) are a class of models that perform language understanding by incorporating a memory component. They perform attention over memory to retrieve all relevant information that may help with the task. In our dialogue modeling setting, we use A_x as the memory component. Our implementation of memory networks, similar to (Bordes and Weston 2016; Dodge et al. 2015), differs from supervised word embeddings described above in only one aspect: how to treat multiple entries in memory. In memory networks, output memory representation $\vec{o} = \sum_i p_i \vec{a}_i$, where $\vec{a}_i$ is the bag-of-words embedding of $a_i \in A_x$ and p_i is the attention signal over memory A_x calculated by $p_i = softmax(\vec{x}^T \vec{a}_i)$. The "compatibility" function of x and y is defined as:

$$f(x, y) = (\vec{x} + \vec{o})^T \vec{y} = \vec{x}^T \vec{y} + (\sum_i p_i \vec{a}_i)^T \vec{y} \quad (8)$$

In contrast to supervised word embeddings described above, attention over memory is determined by message x. This mechanism was originally designed to retrieve information from memory that is relevant to the context, which in our setting is already achieved during commonsense knowledge retrieval. As speculated, the attention over multiple memory entries is better determined by response y in our setting. We empirically prove this point below.

Experiments

Twitter Dialogue Dataset

To the best of our knowledge, there is currently no well-established open-domain response selection benchmark dataset available, although certain Twitter datasets have been used in the response generation setting (Li et al. 2015; 2016). We thus evaluate our method against state-of-the-art approaches in the response selection task on Twitter dialogues.

1.4M Twitter <message, response> pairs are used for our experiments. They were extracted over a 5-month period, from February through July in 2011. 1M Twitter <message, response> pairs are used for training. With the original response as ground truth, we construct 1M <message, response, label=1> triples as positive instances. Another 1M negative instances <message, response, label=0> are constructed by replacing the ground truth response with a random response in the training set.

For tuning and evaluation, we use 20K <message, response> pairs that constitute the validation set (10K) and test set (10K). They are selected by a criterion that encourages interestingness and relevance: both the message and response have to be at least 3 tokens long and contain at least one non-stopword. For every message, at least one concept has to be found in the commonsense knowledge base. For

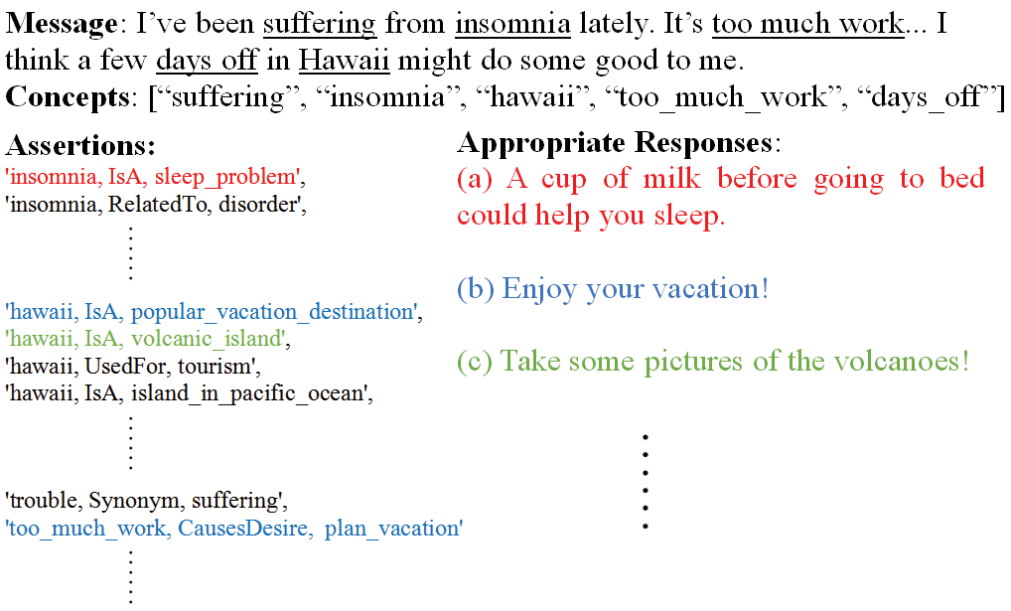

Figure 4: In the illustrated case, five concepts are identified in the message. All assertions associated with the five concepts constitute A_x. We show three appropriate responses for this single message. Each of them is associated with (same color) only one or two commonsense assertions, which is a paradigm in open-domain conversation and provides ground for our max-pooling strategy. It is also possible that an appropriate response is not relevant to any of the common assertions in A_x at all, in which case our method falls back to Dual-LSTM.

each instance, we collect another 9 random responses from elsewhere to constitute the response candidates.

Preprocessing of the dataset includes normalizing hashtags, "@User", URLs, emoticons. Vocabulary V is built out of the training set with 5 as minimum word frequency, containing 62535 words and an extra $<UNK>$ token representing all unknown words.

ConceptNet

In our experiment, ConceptNet[4] is used as the commonsense knowledge base. Preprocessing of this knowledge base involves removing assertions containing non-English characters or any word outside vocabulary V. 1.4M concepts remain. 0.8M concepts are unigrams, 0.43M are bi-grams and the other 0.17M are tri-grams or more. Each concept is associated with an average of 4.3 assertions. More than half of the concepts are associated with only one assertion.

An average of 2.8 concepts can be found in ConceptNet for each message in our Twitter Dialogue Dataset, yielding an average of 150 commonsense assertions (the size of A_x). Unsurprisingly, common concepts with more assertions associated are favored in actual human conversations.

It is worth noting that ConceptNet is also noisy due to uncertainties in the constructing process, where 15.5% of all assertions are considered "false" or "vague" by human evaluators (Speer and Havasi 2012). Our max-pooling strategy used in Tri-LSTM encoder and supervised word embeddings is partly designed to alleviate this weakness.

[4]https://conceptnet.io. ConceptNet can be Downloaded at http://github.com/commonsense/conceptnet5/wiki/Downloads.

Parameter Settings

In all our models excluding term frequency–inverse document frequency (TF-IDF) (Ramos and others 2003), we initialize word embeddings with pretrained GloVe embedding vectors (Pennington, Socher, and Manning 2014). The size of hidden units in LSTM models is set to 256 and the word embedding dimension is 100. We use stochastic gradient descent (SGD) for optimizing with batch size of 64. We fixed training rate at 0.001.

Results and Analysis

The main results for TF-IDF, word embeddings, memory networks and LSTM models are summarized in Table 1. We observe that:

(1) LSTMs perform better at modeling dialogues than word embeddings on our dataset, as shown by the comparison between Tri-LSTM and word embeddings.

(2) Integrating commonsense knowledge into conversational models boosts model performance, as Tri-LSTM outperforms Dual-LSTM by a certain margin.

(3) Max-pooling over all commonsense assertions depending on response y is a better method for utilizing commonsense knowledge than attention over memory in our setting, as demonstrated by the gain of performance of word embeddings over memory networks.

We also analyze samples from the test set to gain an insight on how commonsense knowledge supplements the message itself in response selection by comparing Tri-LSTM encoder and Dual-LSTM encoder.

Table 1: Model evaluation. * indicates models with commonsense knowledge integrated. The TF-IDF model is trained following (Lowe et al. 2015b). The "Recall@k" method is used for evaluation (Lowe et al. 2016b). The model is asked to rank a total of N responses containing one positive response and $N-1$ negative responses ($N=10$ according to our test set). If the ranking of the positive response is not larger than k, Recall@k is positive for that instance.

Recall@k	TF-IDF	Word Embeddings*	Memory Networks*	Dual-LSTM	Tri-LSTM*	Human
Recall@1	32.6%	73.5%	72.1%	73.6%	**77.5%**	87.0%
Recall@2	47.3%	84.0%	83.6%	85.6%	**88.0%**	-
Recall@5	68.0%	95.5%	94.2%	95.9%	**96.6%**	-

Table 2: Case studies for the impact of commonsense assertions. "Activated Assertion" is the commonsense assertion entry in A_x chosen by max-pooling. $\diamondsuit$ indicates correct selection. All 4 instances displayed are taken from the test set.

Instance	Message	Response selected by Dual-LSTM
1	i was helping my brother with his chinese.	did yoga help?
2	bonjour madame, quoi de neuf.	yeah me too !
3	help what colour shoes can i wear with my dress to the wedding?	very pale pink or black.$^\diamondsuit$
4	helping mum paint my bedroom.	shouldn't it be your mum helping you? what color are you going for ?$^\diamondsuit$

Instance	Response selected by Tri-LSTM	Activated Assertion (total size of A_x)
1	the language sounds interesting! i really gotta learn it !$^\diamondsuit$	*chinese, IsA, human_language (755)*
2	loool . you can stick with english , its all good unless you want to improve your french .$^\diamondsuit$	*bonjour, IsA, hello_in_french (9)*
3	very pale pink or black.$^\diamondsuit$	*pink, RelatedTo, colour (1570)*
4	shouldn't it be your mum helping you? what color are you going for ?$^\diamondsuit$	*paint, RelatedTo, household_color (959)*

As illustrated in Table 2, instances 1,2 represent cases where commonsense assertions as an external memory module provide certain clues that the other model failed to capture. For example in instance 2, Tri-LSTM selects the response "...improve your french" to message "bonjour madame" based on a retrieved assertion "$bonjour, IsA, hello\_in\_french$", while Dual-LSTM selects an irrelevant response. Unsurprisingly, Dual-LSTM is also able to select the correct response in some cases where certain commonsense knowledge is necessary, as illustrated in instance 3. Both models select "... pink or black" in response to message "...what color shoes...", even though Dual-LSTM does not have access to a helpful assertion "$pink, RelatedTo, color$".

Informally speaking, such cases suggest that to some extent, Dual-LSTM (models with no memory) is able to encode certain commonsense knowledge in model parameters (e.g., word embeddings) in an implicit way. In other cases, e.g., instance 4, the message itself is enough for the selection of the correct response, where both models do equally well.

Conclusion and Future Work

In this paper, we emphasized the role of memory in conversational models. In the open-domain chit-chat setting, we experimented with commonsense knowledge as external memory and proposed to exploit LSTM to encode commonsense assertions to enhance response selection.

In the other research line of response generation, such knowledge can potentially be used to condition the decoder in favor of more interesting and relevant responses. Although the gains presented by our new method is not spectacular according to Recall@k, our view represents a promising attempt at integrating a large heterogeneous knowledge base that potentially describes the world into conversational models as a memory component.

Our future work includes extending the commonsense knowledge with common (or factual) knowledge, e.g., to extend the knowledge base coverage by linking more named entities to commonsense knowledge concepts (Cambria et al. 2014), and developing a better mechanism for utilizing such knowledge instead of the simple max-pooling scheme

used in this paper. We would also like to explore the memory of the model for multiple message response pairs in a long conversation.

Lastly, we plan to integrate affective knowledge from SenticNet in the dialogue system in order to enhance its emotional intelligence and, hence, achieve a more human-like interaction. The question, after all, is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions (Minsky 2006).

Acknowledgements

We gratefully acknowledge the help of Alan Ritter for sharing the twitter dialogue dataset and the NTU PDCC center for providing computing resources.

References

Bordes, A., and Weston, J. 2016. Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*.

Cambria, E., and Hussain, A. 2015. *Sentic Computing: A Common-Sense-Based Framework for Concept-Level Sentiment Analysis*. Cham, Switzerland: Springer.

Cambria, E.; Hussain, A.; Havasi, C.; and Eckl, C. 2009. Common sense computing: From the society of mind to digital intuition and beyond. In Fierrez, J.; Ortega, J.; Esposito, A.; Drygajlo, A.; and Faundez-Zanuy, M., eds., *Biometric ID Management and Multimodal Communication*, volume 5707 of *Lecture Notes in Computer Science*. Berlin Heidelberg: Springer. 252–259.

Cambria, E.; Hussain, A.; Durrani, T.; Havasi, C.; Eckl, C.; and Munro, J. 2010. Sentic computing for patient centered application. In *IEEE ICSP*, 1279–1282.

Cambria, E.; Song, Y.; Wang, H.; and Howard, N. 2014. Semantic multi-dimensional scaling for open-domain sentiment analysis. *IEEE Intelligent Systems* 29(2):44–51.

Cambria, E.; Poria, S.; Bajpai, R.; and Schuller, B. 2016. SenticNet 4: A semantic resource for sentiment analysis based on conceptual primitives. In *COLING*, 2666–2677.

Dodge, J.; Gane, A.; Zhang, X.; Bordes, A.; Chopra, S.; Miller, A.; Szlam, A.; and Weston, J. 2015. Evaluating prerequisite qualities for learning end-to-end dialog systems. *arXiv preprint arXiv:1511.06931*.

Ghazvininejad, M.; Brockett, C.; Chang, M.; Dolan, B.; Gao, J.; Yih, W.; and Galley, M. 2017. A knowledge-grounded neural conversation model. *CoRR* abs/1702.01932.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Li, J.; Galley, M.; Brockett, C.; Spithourakis, G. P.; Gao, J.; and Dolan, B. 2016. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.

Liu, C.-W.; Lowe, R.; Serban, I. V.; Noseworthy, M.; Charlin, L.; and Pineau, J. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

Lowe, R.; Pow, N.; Charlin, L.; Pineau, J.; and Serban, I. V. 2015a. Incorporating unstructured textual knowledge sources into neural dialogue systems. In *Machine Learning for Spoken Language Understanding and Interaction, NIPS 2015 Workshop*.

Lowe, R.; Pow, N.; Serban, I.; and Pineau, J. 2015b. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. *arXiv preprint arXiv:1506.08909*.

Lowe, R.; Serban, I. V.; Noseworthy, M.; Charlin, L.; and Pineau, J. 2016a. On the evaluation of dialogue systems with next utterance classification. *arXiv preprint arXiv:1605.05414*.

Lowe, R.; Serban, I. V.; Noseworthy, M.; Charlin, L.; and Pineau, J. 2016b. On the evaluation of dialogue systems with next utterance classification. *CoRR* abs/1605.05414.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Minsky, M. 1986. *The Society of Mind*. New York: Simon and Schuster.

Minsky, M. 2006. *The Emotion Machine: Commonsense Thinking, Artificial Intelligence, and the Future of the Human Mind*. New York: Simon & Schuster.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, 1532–1543.

Poria, S.; Cambria, E.; Gelbukh, A.; Bisio, F.; and Hussain, A. 2015. Sentiment data flow analysis by means of dynamic linguistic patterns. *IEEE Computational Intelligence Magazine* 10(4):26–36.

Poria, S.; Chaturvedi, I.; Cambria, E.; and Bisio, F. 2016. Sentic LDA: Improving on LDA with semantic similarity for aspect-based sentiment analysis. In *IJCNN*, 4465–4473.

Rajagopal, D.; Cambria, E.; Olsher, D.; and Kwok, K. 2013. A graph-based approach to commonsense concept extraction and semantic similarity detection. In *Proceedings of the 22nd International Conference on World Wide Web*, 565–570. ACM.

Ramos, J., et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*.

Ritter, A.; Cherry, C.; and Dolan, W. B. 2011. Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural language processing*, 583–593. Association for Computational Linguistics.

Serban, I. V.; Sordoni, A.; Bengio, Y.; Courville, A.; and Pineau, J. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Singh, P.; Lin, T.; Mueller, E.; Lim, G.; Perkins, T.; and Li Zhu, W. 2002. Open mind common sense: Knowledge

acquisition from the general public. *On the move to meaningful internet systems 2002: CoopIS, DOA, and ODBASE* 1223–1237.

Speer, R., and Havasi, C. 2012. ConceptNet 5: A large semantic network for relational knowledge. In Hovy, E.; Johnson, M.; and Hirst, G., eds., *Theory and Applications of Natural Language Processing*. Springer. chapter 6.

Sukhbaatar, S.; Weston, J.; Fergus, R.; et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, 2440–2448.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.

Vinyals, O., and Le, Q. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Wang, Q.; Cambria, E.; Liu, C.; and Hussain, A. 2013. Common sense knowledge for handwritten chinese recognition. *Cognitive Computation* 5(2):234–242.

Weston, J.; Bordes, A.; Chopra, S.; Rush, A. M.; van Merriënboer, B.; Joulin, A.; and Mikolov, T. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.

Weston, J.; Chopra, S.; and Bordes, A. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

Xing, C.; Wu, W.; Wu, Y.; Liu, J.; Huang, Y.; Zhou, M.; and Ma, W. 2016. Topic augmented neural response generation with a joint attention mechanism. *CoRR* abs/1606.08340.

Zhou, X.; Dong, D.; Wu, H.; Zhao, S.; Yan, R.; Yu, D.; Liu, X.; and Tian, H. 2016. Multi-view response selection for human-computer conversation. *EMNLP'16*.