The MSR-NLP System at Dialog System Technology Challenges 6

Michel Galley, Chris Brockett, Bill Dolan, and Jianfeng Gao

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052, USA
{mgalley,chrisbkt,billdol,jfgao}@microsoft.com

Abstract

We present our work on the Dialog System Technology Challenges 6 (DSTC6). We participated in Track 2, which evaluates the generation of conversational responses in a fully data-driven manner. Our system follows the approach taken by Li et al. [1], which utilizes sequence-to-sequence (SEQ2SEQ) models that exploit a Maximum Mutual Information (MMI) criterion that has been shown to increase response adequacy and diversity. We find that when trained on the DSTC6 corpus MMI models exhibit improvements in BLEU scores, CIDEr, and SkipThoughts over the task baseline, but not METEOR or ROUGE-L. We also show gains in terms of unigram and bigram lexical diversity. However, inspection of the datasets used in the DSTC6 Track 2 task suggests that the task may favor blander outputs. In particular, the high incidence of references to taking the conversation offline suggests that the datasets may skewed to favor a single response type.

Index Terms: dialog, conversation models, deep learning, LSTM, mutual information, MMI

1. Introduction

The Dialog System Technology Challenges[1] (DSTC) in its sixth edition offers for the first time a track (Track 2) [2] devoted exclusively to fully data-driven approaches to building dialog systems. The MSR-NLP entry to Track 2 abides to this constraint, and was trained exclusively from conversational data and does not make use of any rule-based or hand-coded component.[2]

Our approach for this track is based almost entirely on [1], which is summarized in Section 2. Our five entries to this track use LSTM SEQ2SEQ models, and exploit a maximum mutual information (MMI) criterion at decoding time. This use of MMI was shown in [1] to promote responses of improved quality (according to BLEU and human assessments) and of greater lexical diversity. Our participation in this track allows us to explore the utility of MMI in the context of a more task-oriented system than the more chitchat-oriented dialog to which it has hitherto been applied.

This report explores different parameterizations of LSTM SEQ2SEQ models, of the MMI criterion, and different decoding hyperparameters (Section 4). It also analyzes the difference between findings of [1] on general Twitter conversation data, and our findings on the DSTC6 Track 2 dataset. The main difference is that the former is a free-form, very open domain dataset of mostly chit-chat conversations, while the latter is of somewhat narrower domain (mainly customer support) and constitutes mostly of task-oriented dialog. Our main finding has to do with the MMI objective, which helps avoid deflective and bland responses. Whereas MMI helped promote more diverse, interesting, and engaging dialog in [1], this diversification yields responses that are more risky and therefore potentially undesirable in the more formal and task-oriented setting of DSTC6. We also observe that the kind of deflective responses particular to the DSTC6 dataset (e.g., *please DM us and we will try our best to help*) are often completely appropriate due to user privacy concerns, but only postpone rather than resolve anything in the underlying task. Based on this general observation, we recommend some changes in the end-to-end training track for DSTC7, and make suggestions along this line in Section 5.

2. Models

Our system is based on the end-to-end training approach of [1], and this section notes along the way differences with that paper.

LSTM SEQ2SEQ model. The basic model architecture is an LSTM-to-LSTM model without attention [3].[3] We describe it here in mathematical details for completeness and as to make extensions (MMI, etc.) clearer. Given a sequence of input words (dialog history) $S = \{s_1, s_2, ..., s_{N_s}\}$, the LSTM associates each time step k with an input gate, a memory gate, and an output gate, denoted respectively as i_k, f_k and o_k. N_s represent the number of words in S. We distinguish e and h where e_k is the embedding vector for an individual word at time step k, and h_k is the vector computed by the LSTM model at time k by combining e_k and h_{k-1}. c_k is the cell state vector at time k, and σ represents the sigmoid function. Then, the hidden state h_k for each time step k is given by:

$$i_k = \sigma(W_i \cdot [h_{k-1}, e_k]) \quad (1)$$
$$f_k = \sigma(W_f \cdot [h_{k-1}, e_k]) \quad (2)$$
$$o_k = \sigma(W_o \cdot [h_{k-1}, e_k]) \quad (3)$$
$$l_k = \tanh(W_l \cdot [h_{k-1}, e_k]) \quad (4)$$
$$c_k = f_k \cdot c_{k-1} + i_k \cdot l_k \quad (5)$$
$$h_k^s = o_k \cdot \tanh(c_k) \quad (6)$$

where $W_i, W_f, W_o, W_l \in \mathbb{R}^{D \times 2D}$. In this response generation task, each conversational context S is paired with a sequence of output words to predict: $T = \{t_1, t_2, ..., t_{N_t}\}$. N_t is the length of the response (terminated by an *EOS* symbol) and t represents

[1] Formerly known as "Dialog State Tracking Challenge".

[2] We also refrained from performing any rule-based pre- and post-processing, even though error analysis on a development set suggested we could have prevented relatively common errors (e.g., repeated words) using simple post-processing rules. We feel such processing would run against the spirit of the track.

[3] As in [1], we did not use the attention model [4] as it only gave marginal gains at the expense of significantly longer training time. Other forms of attention were shown to be more suitable for dialogue generation [5], but we did not experiment with the latter for DSTC6.

a word token that is associated with a D-dimensional word embedding e_t (distinct from the source). The LSTM model defines a distribution over output words and sequentially predicts each token using the softmax function:

$$p(T|S) = \prod_{k=1}^{N_t} p(t_k|s_1, s_2, ..., s_t, t_1, t_2, ..., t_{k-1}) = \prod_{k=1}^{N_t} \frac{\exp(f(h_{k-1}, e_{y_k}))}{\sum_{y'} \exp(f(h_{k-1}, e_{y'}))}$$

where $f(h_{k-1}, e_{y_k})$ is the activation function between h_{k-1} and e_{y_k}, where h_{k-1} is the output hidden vector at time $k-1$. Each sentence concludes with a special end-of-sentence marker *EOS*. As it is common, input and output use different LSTMs with separate parameters to capture different patterns of word composition.

Maximum Mutual Information. The standard objective function for SEQ2SEQ models is the log-likelihood of the target T given the source S, which at decoding time yields this statistical decision problem:

$$\hat{T} = \arg\max_T \{ \log p(T|S) \} \tag{7}$$

This formulation often leads to generic and safe responses, since it only selects for targets given sources, not the converse. To mitigate this problem, we replace it with Maximum Mutual Information (MMI) as the objective. In MMI, parameters are chosen to maximize (pairwise) mutual information between source S and target T:

$$\log \frac{p(S,T)}{p(S)p(T)} \tag{8}$$

This avoids producing responses that unconditionally enjoy high likelihood, and instead biases the system towards responses that are specific to the current conversation. The MMI objective can written as follows:

$$\hat{T} = \arg\max_T \{ \log p(T|S) - \log p(T) \}$$

[1] generalized the MMI criterion with a hyperparameter λ that controls how much to penalize generic responses:

$$\hat{T} = \arg\max_T \{ \log p(T|S) - \lambda \log p(T) \} \tag{9}$$

An alternate formulation of the MMI objective exploits Bayes' theorem:

$$\log p(T) = \log p(T|S) + \log p(S) - \log p(S|T)$$

As in [1], this lets us re-write Equation 9 as follows:

$$\begin{aligned} \hat{T} &= \arg\max_T \{ (1-\lambda) \log p(T|S) \\ &\quad + \lambda \log p(S|T) - \lambda \log p(S) \} \\ &= \arg\max_T \{ (1-\lambda) \log p(T|S) + \lambda \log p(S|T) \} \end{aligned} \tag{10}$$

This weighting of the MMI objective can thus be viewed as introducing a tradeoff between source given target (i.e., $p(S|T)$) and target given source (i.e., $p(T|S)$).

For reasons explained in [1], we did not train our models directly using the MMI objective. Instead, we trained separate maximum likelihood models, and used the MMI criterion only during testing, as explained in the remainder of this section.

	Dialogs	Utterances	Words
Train	887,984	2,156k	39,794k
Dev	107,474	262k	4,868k
Dev500	500	1319	24,760
Test	2,000	5266	-

Table 1: DSTC Track 2 dataset statistics

Training. Responses can be generated either from Equation 9, i.e., $\log p(T|S) - \lambda \log p(T)$ or Equation 10, i.e., $(1-\lambda) \log p(T|S) + \lambda \log p(S|T)$. For DSTC6, we only make use of the latter one, which we refer to as MMI-bidi. Direct optimization of MMI-bidi is intractable, as the second term (i.e., $p(S|T)$) requires completion of response generation *before* $p(S|T)$ can effectively be computed. Due to the exponential search space of target sequences T, exploring all possibilities is infeasible. For practical reasons, therefore, we turn to an approximation that involves first generating N-best lists given the first part of objective function, i.e., the standard SEQ2SEQ model $p(T|S)$. Then, we re-rank this N-best lists using the second term of the objective function. Since N-best lists produced by SEQ2SEQ models are usually grammatical, the final selected response is likely to be well-formed as well. Model reranking has obvious drawbacks. It results in not globally optimal solutions by emphasizing standard SEQ2SEQ objectives. Moreover, it relies heavily on the system's success in generating a sufficiently diverse N-best list, requiring that a large N-best list be generated for each message. Nonetheless, this MMI criterion works well in practice, significantly improving both in terms of interestingness and diversity.

Practical considerations. Research has shown that deep LSTMs work better than single-layer ones for SEQ2SEQ tasks [6, 3]. We selected a deep structure with three LSTM layers for encoding and also three LSTM layers for the decoder, each LSTM consisting of a different set of parameters. Each LSTM layer consists of 500 hidden units, and the dimensionality of embeddings vectors is set to 500. Other training details are given below, generally aligned with [3]. LSTM model parameters and embeddings are initialized from a uniform distribution in $[-0.08, 0.08]$. We used stochastic gradient decent (SGD) with a fixed learning rate of 0.1, a batch size of 256, and we clipped gradients, scaling gradients when the norm exceeded a threshold of 1. The $p(S|T)$ model described was trained using the same model as that of $p(T|S)$, with messages (S) and responses (T) interchanged. Note that the capacity of our DSTC model is lower than that of [1], as the training set here is more than an order of magnitude smaller ([1] used 4 layers and 1000-dimensional hidden vectors and embeddings).

Decoding We generate N-best lists using our $p(T|S)$ baseline model, and then rerank this list by linearly combining $\log p(T|S)$, $\lambda \log p(S|T)$, and γN_t. N_t is the number of words of the response, and its parameter γ lets us control the average length of system responses. We used MERT [7] to tune the weights λ and γ on the Dev500 set.[4]

[4]We could have used grid search instead of MERT, as there are only 3 features and 2 free parameters. In either case, the optimizer attempts to find the best tradeoff between $p(T|S)$ and $p(S|T)$ according to BLEU (which tends to weight the two models relatively equally) and ensures that generated responses are of reasonable length.

System	Diversity unigram	Diversity bigram	BLEU B-1	BLEU B-2	BLEU B-3	BLEU B-4	METEOR	ROUGE-L	CIDEr	Skip-Thoughts
baseline	1.13	3.74	23.02	10.93	6.37	3.89	**12.27**	**19.67**	18.69	46.14
MSR-baseline	2.48	8.51	20.51	8.47	4.64	2.85	9.36	16.19	16.41	43.00
MSR-MMI-uniform	3.47	13.99	21.71	9.77	5.66	3.60	9.96	17.67	21.30	44.34
MSR-MMI-maxBLEU	3.13	12.42	**23.63**	**11.09**	**6.50**	**4.12**	10.75	19.17	**21.99**	**46.35**
MSR-MMI-mixed	4.01	16.28	22.66	10.51	6.17	3.95	10.41	18.09	20.46	43.63
MSR-MMI-maxdiv	**4.23**	**17.60**	21.76	10.29	6.14	4.05	10.16	18.06	21.96	44.57
Gold responses	8.98	38.50								

Table 2: 1-reference results (percentages) for baseline and MSR-NLP submissions (Team 5). All results are official ones provided directly by the organizers, except diversity metric (unigram and bigram diversity) which we computed ourselves.

3. Data

The main task for Track 2 of DSTC6 consists of training an end-to-end system from Twitter. It also offered a pilot task based on OpenSubtitles data, but we did not participate in this pilot. Unlike some earlier work exploiting social media data [8, 9], the dataset for Track 2 was purposely restricted to model conversational responses only of customer support Twitter users. This is motivated by a desire to move away from chitchat dialog and aim for more "useful" (e.g., informational) exchanges, following an earlier attempt [10] to generate more informational responses.[5]

We downloaded Track-2 data through the Twitter API using the scripts provided by the organizers.[6] Table 1 summarizes the Twitter data downloaded and generated for this track. Dialog durations ranged between 2 and 20-turns, with a large percentage of dialogs being 2-turn conversations (79%). Each conversation is a back-and-forth between a user (U) and customer support (S), with each dialog ending with a customer support response (S-response, henceforth). The actual generation task is to generate this final S-response based on the previous conversation history.

We used the Train and Dev500 datasets to train our system as explained in the previous section.[7] We trained $p(T|S)$ and $p(S|T)$ models using the Train dataset, and we used Dev500 to tune the two hyperparameters (λ and γ) using MERT.[8] We did not require use of the Dev dataset. Nevertheless, we did not merge Dev into the training data, as we wanted our system to be comparable to the official baseline which also didn't train on Dev. For each dialog of the test set, we generated the final S-response, i.e., 2k responses in total for each of our submissions.

We note finally that Track 2 constitutes an experimental condition quite divergent from what the MSR-NLP system was designed for in our previous research [1, 11, 10]. Data for this track contains fewer than a million dialogs, while the MSR-NLP research system has been trained with up to 140 million conversations in order to model free-form and open-domain dialog.

[5] We note an important difference between Track 2 and the experimental setting of [10]: Track 2 exclusively models customer support responses, while [10] deliberately removed customer support from the knowledge-grounded conversations, as customer support tends to take conversations offline (i.e., direct messages).

[6] A script provided by the organizer performed a sanity check of our download compared to the organizers' own training data. We found that only 0.02% of the dialogs, 0.02% of the utterances, and 0.68% of the words were different from the gold standard. These percentages fall below the 1% threshold which would have required us to contact the track organizers.

[7] The split into Train, Dev, Dev500, Test was done by the organizers.

[8] We tuned on Dev500, as tuning on Dev would have been too slow.

4. Experiments

We evaluate our submissions and the baseline using corpus-level BLEU [12] (BLEU1 to BLEU4), CIDEr [13], ROUGE-L [14], and SkipThoughts [15]. Results for these metrics are provided by the DSTC organizers themselves. In addition, we evaluate the lexical diversity of the systems using distinct-1 and distinct-2 [1], which compute at the corpus-level the number of unique 1-grams (or 2-grams) divided by the total number of 1-grams (or 2-grams) generated by the system.

We submitted five systems to Track 2, representing different tradeoffs between accuracy and diversity. We compare these systems to the baseline provided by the organizers, which is a reimplementation of [16]. These five submissions are:

- **MSR-baseline**: A vanilla LSTM SEQ2SEQ model without MMI, which correspond to the baseline system in [1]. It uses greedy search—which typically leads to more search errors than beam search—as a way of increasing response diversity.
- **MSR-MMI-uniform**: The system of [1] (MMI-bidi), without hyperparameter tuning. Model scores $\log p(T|S)$ and $\log p(S|T)$ are normalized by sentence length, and the reranking step does not add any word penalty or bonus to affect response length.
- **MSR-MMI-maxBLEU**: The system of [1] (MMI-bidi), with hyperparameter tuning using MERT. We used 50 runs of MERT from random starting points.
- **MSR-MMI-maxdiv**: A variant of MMI-maxBLEU, targeting greater diversity. We ran 50 distinct runs of MERT from different random initializations, and selected the one with the highest unigram diversity.
- **MSR-MMI-mixed**: A variant of MMI-maxBLEU, targeting a balance between diversity and BLEU. We ran 50 distinct runs of MERT from different random initializations, and selected the one with highest weighted score (BLEU + unigram diversity).

The main automatic evaluation results with 1 reference are shown in Table 2. The different versions of MMI show improvements over the baseline in terms of BLEU, CIDEr and SkipThought scores. As expected, the system trained to maximize BLEU (MSR-MMI-maxBLEU) on the dev set reaches the highest BLEU scores on the test set. The more striking difference between the baseline and all our systems is in terms of unigram and bigram diversity. While these measures are not

U: shout out to whoever in my building keeps stealing my @1800petmeds orders . fighting heartworm one theft at a time .
S: very sorry to hear that ! do you have your order now , or can we assist you ?
U: appreciate your help ! o <NUMBERS> - will have orders sent to my office from now on ...
S: can you dm us the address to which we should send the replacement ? thank you ! <URL>

Table 3: Taking the conversation offline: Here, there is clear motivation to maintain the user's privacy.

System	Div. 1-gram	BLEU B-1	B-2	B-3	B-4	METEOR	ROUGE-L	CIDEr	Skip-Thoughts	Human
MSR-baseline	2.48	51.11	27.28	16.03	9.91	16.87	31.46	7.08	59.52	3.3054
MSR-MMI-uniform	3.47	54.80	31.15	19.37	12.61	17.54	33.10	9.45	61.51	3.4546
MSR-MMI-maxBLEU	3.13	**59.25**	**36.27**	**23.64**	**15.75**	**19.18**	**36.58**	**11.12**	**64.57**	3.5098
MSR-MMI-mixed	4.01	57.90	34.30	21.91	14.48	18.39	33.75	9.40	60.25	**3.5396**
MSR-MMI-maxdiv	**4.23**	58.14	34.22	21.98	14.80	18.13	33.88	10.25	61.31	3.5209

Table 4: Multi-reference results (percentages) for baseline and MSR-NLP submissions, and human evaluation (Team 5). All results (other than 1-gram diversity) are official ones.

evaluation metrics *per se*, this at least indicates that our systems are conditioned to avoid catch-all responses such as *"please DM us and we will help you"*. The most diverse of our systems produces a unigram diversity of 4.23% on the test set, which is about half of the diversity on gold-standard responses (8.98%). Achieving good diversity on this data is a challenge, as it is lexically quite homogeneous.[9] CIDEr scores are mostly consistent with those of BLEU, which is not surprising as CIDEr was meant to mitigate the weaknesses of BLEU when the number of references is particularly large (e.g., 50 or more, much more than what is available for DSTC6).

On the negative side, the official baseline system outperforms all our submissions in terms of METEOR and ROUGE-L scores. ROUGE is a recall oriented metric designed for summarization, while METEOR is a translation metric with tunable hyperparameters that were specifically optimized for translation and not conversation. Both metrics tend to favor longer responses [17], which may partially explain the good performance of the baseline (baseline system output is on average 15.1% longer than the reference).

The results on METEOR are strikingly different from those of other metrics, so we further analyzed our poor performance on METEOR. Table 5 suggests that this metric is not particularly suited for this DSTC task. Indeed, as the test set is single reference, we computed both METEOR of the hypotheses against the references and of the references against the hypotheses. We believe that the outcomes of the two approaches should ideally be consistent, as these metrics are meant to measure the degree of semantic or pragmatic equivalence between two responses. Equivalence is obviously a commutative relation (i.e., $x = y \implies y = x$). Table 5 shows this consistency emerging with BLEU but not METEOR. Accordingly, we feel it is reasonable to discount our results on METEOR as less reliable.[10]

Finally, results for multi-reference automatic evaluation and human evaluation are shown in Table 4. There are 11 references, i.e., the original reference plus 10 references crowdsourced by the conference organizers. Results are relatively consistent with those of Table 2, with MSR-MMI-maxBLEU on top in terms of BLEU and other metrics. Human scores are relatively consistent with automatic scores, except for MSR-MMI-mixed and MSR-MMI-maxdiv which humans rated more favorably. We think this discrepancy is again an artifact of the data and task, as a large percentage of responses are of the form *please DM us*, which amount to almost no information in this context. Hence, humans seemed to prefer responses that attempted to steer away from these commonplace responses. The best system according to human evaluation (MSR-MMI-mixed) is one that balances response adequacy (as approximated by BLEU) and diversity.

	baseline	MMI-maxBLEU
BLEU(ref,hyp)	3.76	**4.12**
BLEU(hyp,ref)	3.74	**4.09**
METEOR(ref,hyp)	**11.79**	10.73
METEOR(hyp,ref)	10.42	**11.22**

Table 5: Relative symmetricity of BLEU-4, and lack thereof for METEOR. The results for the baseline differ slightly from Table 2, as we used here a baseline model trained ourselves using the organizer's implementation (we didn't have their models or output to conduct these extra experiments).

5. Discussion

In the course of our experiments, we observed that training data set appears to be inherently biased towards certain customer service interactions characteristic of public online forums. Of particular note is the fact that 172,489 of the 887,984 responses (19.4%) in the training set contain the string " dm " ("direct message"). Altogether, approximately 25.1% of the responses in the training data contain some reference to "message", "e-mail", "email" or "dm".[11] Manual analysis of a random sample of these matches suggest they largely represent attempts by customer support representatives to take the discussion offline. In this respect, the Task 2 dataset may not be be an ideal dataset with which to exercise algorithmic alternatives.

[9] As a point of comparison, a random sample of Twitter responses of the same size as DSTC test set – i.e., of 2000 turns – gave us unigram and bigram diversities of 16.2% and 64.4%, i.e., almost twice as high.

[10] This is not critique of METEOR in general, as this inconsistency may be due to METEOR hyperparameters being tuned specifically on translation data, typically with more than one reference.

[11] By way of contrast, [1] motivate their use of MMI by observing that in the OpenSubtitles database, 0.45% sentences contain the sequence *I dont know*.

We observe that the baseline system ([3]) generates a high proportion of responses along the lines of *please DM the details and we will follow up*. Some 51.2% of the baseline responses contain one of *DM us*, *DM me*, *please DM* or *could DM*. Given the requirements of customer privacy (exemplified in Fig. 3), this is a perfectly reasonable thing to do. Overall, 20.8% of reference test set responses, 63.6% of baseline system outputs, and 37.4% of our maxBLEU system outputs contained the string "DM". It is evident from this that MMI–originally designed to prevent commonplace responses such as *I don't know* and *sounds like a plan* in chitchat dialogs–increases diversity of the generated output over the baseline on the Task 2 dataset, it may be less well-suited to a customer service setting where bland ("safe") responses may be more appropriate than the more "interesting" responses that MMI promotes. Many of our outputs seem felicitous in the right context, e.g., *your order is in the shipping process and should be with you in 4-5 business days*, but others would be highly implausible in any context, e.g., *another unhappy customer. thanks for sharing. this helps other customers make better purchasing decisions.*.

We note also the prevalence of simple acknowledgements and expressions of appreciation in the responses in the training set such as *thanks for pointing this out*, *thanks for the feedback*, *thanks for the shout out*, and *glad you like it*. A simple grep for several varieties of smiley emoticons and emoji, together with the strings *re welcome* and *great to hear* found these forms in 6.2% of responses, suggesting that in a significant subset of cases, the problem may have been resolved before the final turn (e.g., *great to hear* suggests that the user figured out a solution, or that the underling problem was solved).

These observations lead us to believe that the data used in the Task 2 do not fully represent the challenges of customer service agents, and that future tasks, while remaining conversational and data-driven, should be either more goal *completion*-oriented, or, drawing on in-domain side data as external resources, information-oriented [10].

6. Conclusions

DSTC6 Task 2 provided a valuable opportunity to investigate the possibility of fully data-driven conversation in a more goal-oriented scenario than the casual chitchat that we have focused on in the past. It also furnished a useful framework in which to explore the applicability of MMI when trained and tested on Twitter customer service exchanges, and to calibrate our system with others on the basis of a shared dataset. We found that MMI did improve BLEU and diversity metrics over the baseline system, and over our own "vanilla" Sequence-to-Sequence model. However, it is not entirely clear that the MMI models are suited to the task as presently formulated, inasmuch as dataset itself is intrinsically skewed towards a particular type of commonplace response in which the communication is taken offline.

We recommend that future tasks might be either more explicitly goal-completion-oriented or more side-information-oriented in order to mitigate some of the issues that became apparent in the course of performing this DSTC6 task.

7. References

[1] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2016, pp. 110–119.

[2] C. Hori and T. Hori, "End-to-end conversation modeling track in DSTC6," *arXiv:1706.07440*, 2017.

[3] I. Sutskever, O. Vinyals, and Q. Le, "Sequence to sequence learning with neural networks," in *Proc. of NIPS*, 2014, pp. 3104–3112.

[4] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2014.

[5] L. Shao, S. Gouws, D. Britz, A. Goldie, B. Strope, and R. Kurzweil, "Generating long and diverse responses with neural conversation models," *CoRR*, vol. abs/1701.03185, 2017. [Online]. Available: http://arxiv.org/abs/1701.03185

[6] O. Vinyals, L. Kaiser, T. Koo, S. Petrov, I. Sutskever, and G. Hinton, "Grammar as a foreign language," in *Proc. of NIPS*, 2015.

[7] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan: Association for Computational Linguistics, July 2003, pp. 160–167. [Online]. Available: http://www.aclweb.org/anthology/P03-1021

[8] A. Ritter, C. Cherry, and W. Dolan, "Data-driven response generation in social media," in *Proc. of EMNLP*, 2011, pp. 583–593.

[9] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan, "A neural network approach to context-sensitive generation of conversational responses," in *Proc. of NAACL-HLT*, May–June 2015.

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

[11] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A persona-based neural conversation model," *ACL*, 2016.

[12] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. of ACL*, 2002.

[13] R. Vedantam, C. L. Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 4566–4575.

[14] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, S. S. Marie-Francine Moens, Ed. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81.

[15] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, "Skip-thought vectors," *CoRR*, vol. abs/1506.06726, 2015.

[16] O. Vinyals and Q. Le, "A neural conversational model," in *Proc. of ICML Deep Learning Workshop*, 2015.

[17] F. Guzmán, P. Nakov, and S. Vogel, "Analyzing optimization for statistical machine translation: Mert learns verbosity, pro learns length," in *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, Beijing, China, July 2015, pp. 62–72.