Augmenting Memory Networks for Rich and Efficient Retrieval in Grounded Dialogue

Anonymous ACL submission

Abstract

Grounded dialogue consists of conditioning a conversation on additional latent inputs ("factoids") beyond the dialogue context, such as Wikipedia articles, IMDB reviews, persona, and images. Due to a scarcity of <context, factoid> labels, it is common practice to jointly learn the knowledge-selection and grounded response generation tasks end-to-end. When conditioning the response on these factoids, previous work has either treated the factoids as a weighed average vector, or separately computed probabilities for each <context, factoid> pair. However, the former creates a bottleneck whilst the latter prevents factoids from being considered jointly. Our new method, PolyMemNet, learns a matrix representation of the context and factoids, allowing for multiple factoids to be jointly considered in response selection, without imposing a bottleneck. We show how this achieves up to a 17% boost in knowledge-selection accuracy and 13% in response-selection accuracy versus memory networks.

1 Introduction

There is growing interest in grounded dialogue models which can condition their responses on additional latent inputs ("factoids") beyond the dialogue context, such as Wikipedia articles (Dinan et al., 2019), IMDB reviews (Moghe et al., 2018), persona (Zhang et al., 2018), and images (Mostafazadeh et al., 2017). This allows the model to access latent information implicitly assumed by speakers. It is an effective means of reducing hallucination (Shuster et al., 2021) whereby the model produces plausible but factually inaccurate responses and genericness whereby the model produces mainly common but bland responses (e.g. "I think so", "that's right" etc.) (Li et al., 2016). Following Dinan et al. (2019), we are concerned principally with the tasks of knowledge-selection, choosing an appropriate factoid given the dialogue context,

Context	[A] Hi how are you doing? I am okay how about you? [B] I used to do home health aide but now I am disabled .
Persona	I love to drink fancy tea. I have a big library at home. I'm a museum tour guide. I'm partly deaf.
Ground truth	I am sorry to hear that. What happened
MemNet	I currently work for a museum.
PolyMemNet (ours)	That is no good. I'm deaf so it limits me to what I can do.

Table 1: **Example predictions from the Persona Chat validation set.** PolyMemNet successfully chooses a response that both incorporates persona and responds to the dialogue context, while MemNet ignores the context and just repeats the persona almost verbatim.

and response-selection, choosing an appropriate response given both the context and chosen factoid (table 1). Learning the knowledge-selection function through supervision (Kim et al., 2020; Dinan et al., 2019) scales poorly due to the scarcity of <context, factoid> labels, which require manual annotation by crowdworkers. Generating pseudo-labels through <factoid, response> similarity automates this, but requires careful engineering of the similarity function which may not generalise to different types of grounding where there is limited surface-level semantic similarity. For these reasons, it is common practice to jointly learn both tasks end-to-end (i.e. unsupervised knowledge-selection).

The requirement for gradients to flow between knowledge-selection and response-selection modules materially constrains how the factoids can interact with the context during the response-selection phase. Specifically, in previous works, factoids are either treated as a weighted-average vector (Mazaré et al., 2018; Fan et al., 2021), or separate probability distributions are computed for each <context, factoid> pair, before a final marginalisation step (Bruyn et al., 2020; Shuster et al., 2021; Zhang et al., 2021). Both approaches have significant downsides: the former creates a bottleneck similar to RNNs as the information from multiple factoids must be compressed into a single vector (Bahdanau et al., 2015) and also makes it difficult to represent the one-to-many relation between contexts and factoids (Kim et al., 2020); the latter prevents factoids from interacting with each other and is much more memory intensive. This is especially problematic for retrieval models, because performance often scales with batch size (Humeau et al., 2020).

We present an architecture, PolyMemNet, that removes this bottleneck whilst allowing factoids to be considered jointly when selecting a response. Specifically, our model extends the memory network architecture (Sukhbaatar et al., 2015) by learning multiple latent vector representations of the dialogue context which separately attend to the factoids, thereby allowing a rich interaction between context, factoids, and response candidates (table 1). We show how this achieves up to a 17% boost in knowledge-selection accuracy and 13% in response-selection accuracy, while maintaining a similar memory footprint to memory networks. We also show how this end-to-end method offers enhanced generalisation over unseen topics compared to supervised models. Our key contributions are:

- Removing the single vector bottleneck to enable **multiple hypotheses** to be jointly considered in the response-selection process.
- Learning knowledge-selection in an **unsupervised** manner, allowing for arbitrary sources of grounding such as **documents** or **persona**.
- A **memory-efficient** solution in which performance scales with number of latent vectors without material increases in memory usage.

We anonymously make our code publicly available on GitHub to enable reproducibility[1].

2 Related work

Early work in grounded NLP separated the task into knowledge-selection and response/answer-generation (Dinan et al., 2019). Typically, coarse-grained sparse retrieval techniques such as BM25 (Robertson and Zaragoza, 2009) and

[1] github.com/AtticRuckverwandlung/AugmentingMemoryNetworks

TF-IDF (Ramos, 2003) were used to reduce the search space over factoids, followed by a neural reranker to perform fine-grained evaluation over candidates (Chen et al., 2017). Recent approaches learn the entire process end-to-end (Lian et al., 2019), often multitasking on labelled knowledge-selection data to reduce the noisiness of retrieved factoids in the early part of training (Dinan et al., 2019; Kim et al., 2020).

Two contrasting approaches to context-factoid interaction in the response generation phase have emerged: a) Vector-based approaches obtain a weighted-average factoid vector which is summed/concatenated with the context representation (Mazaré et al., 2018; Fan et al., 2021) or employ differentiable sampling techniques such as Gumbel-Softmax (Jang et al., 2017; Lian et al., 2019); b) Marginalisation approaches compute separate probability scores $p(y|x, z_k)$ for each <context, factoid> pair and marginalise only at the end (Shuster et al., 2021; Zhang et al., 2021; Bruyn et al., 2020). The former allows factoids to be considered jointly, but bottlenecks them into a single vector. The latter removes the bottleneck, but is more memory-intensive and cannot consider factoids jointly.

Our PolyMemNet model takes inspiration from late-stage interaction techniques such as the Polyencoder in response retrieval (Humeau et al., 2020) and ColBERT in openQA (Khattab et al., 2021), which have bridged the performance-gap between *bi-encoders*, in which contexts and responses only interact via a final dot product score and *cross-encoders*, which perform full all-on-all attention. Our approach differs in that we use the latent codes as a hidden state which accumulates information from both the context and factoids to perform grounded response-selection, rather than simply in a paired retrieval task such as <context, response> or <query, document> retrieval.

3 Methodology

PolyMemNet (figure 1) comprises context representation (CR), knowledge-selection (KS) and response-selection (RS) modules. The CR module learns multiple vector representations of the context, which in the KS module separately query different information from the factoids. In the RS module, it applies pseudo-relevance feedback (Cao et al., 2008) with the response candidates on the joint <context, factoids> representation, before obtaining the final <context, response> scores.

3.1 Memory Networks

In the standard memory network ("MemNet") (see figure 1) (Sukhbaatar et al., 2015), we obtain a vector representation of the context $\mathbf{x} = enc(x)$ as a query and then perform dot product attention with a set of memory vectors $\mathbf{H_z}$ (i.e. the factoids) as keys and values to obtain a weighted average representation of them. We then re-add the initial query via a skip-connection to obtain a joint representation $\mathbf{o_z}$. Lastly we define the scoring function $S(x, y)$ as the dot product between $\mathbf{o_z}$ and the vector representation of the response $\mathbf{y}$ and train the model via cross entropy loss, using the other samples in the batch as negatives.

3.2 Context representation module

MemNet can be seen as a special case of a class of models that operate over sets of latent vectors ("codes"). We begin with some randomly initialised learnable latent codes $\mathbf{C} \in \mathbb{R}^{l \times d}$ where l is the number of latent codes (we use $l = 128$ for our default setup; see table 3) and d is the dimensionality of the model. We define the context representation $\mathbf{O_x}$ as the result of dot product attention between these codes as queries and the context embeddings $\mathbf{H_x} \in \mathbb{R}^{n \times d}$ as keys and

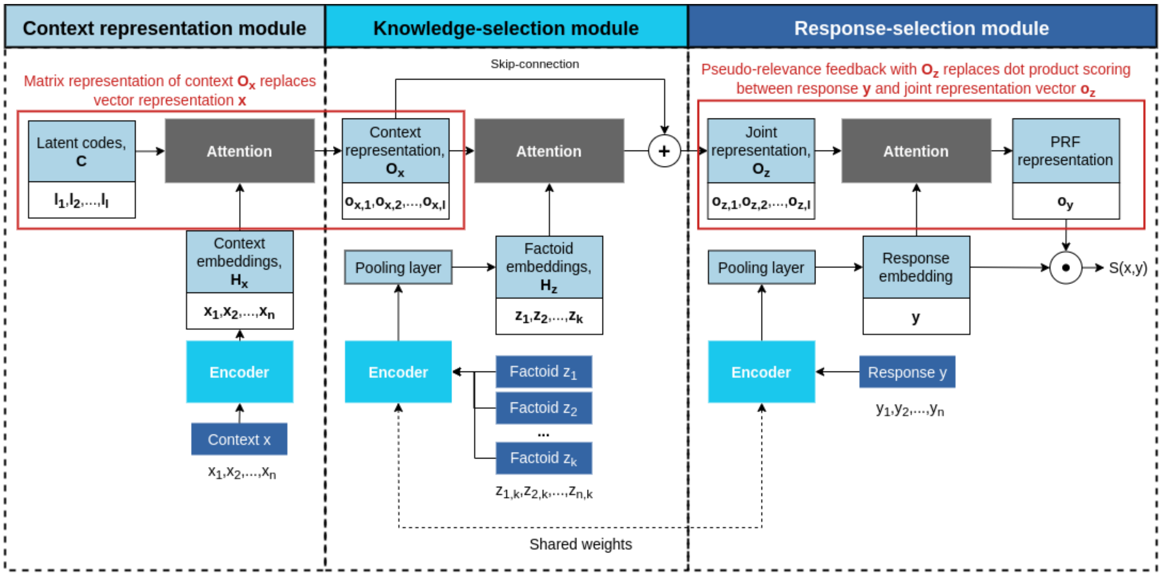

Figure 1: The **PolyMemNet** architecture. $\odot$ denotes a dot product function and $\oplus$ denotes element-wise addition. (red) highlights differences with the standard MemNet model.

values:

$$\mathbf{O_x} = softmax(\mathbf{C}\mathbf{H_x^T})\mathbf{H_x}$$

Note, the only difference between PolyMemNet and MemNet when $l = 1$ is the use of a learned code to extract a linear combination of context embeddings, rather than the [CLS] token or mean pooling.

3.3 Knowledge-selection module

For a given context, there are many plausible factoids which could be used to generate a response. This one-to-many relation (Kim et al., 2020) makes learning to select a factoid based on a single context vector difficult, as it is pulled in different directions by competing factoids. By using multiple latent context vectors however, each vector can learn a specialisation, similar to the effect of multi-headed attention in transformers (Voita et al., 2019). This allows the model to 'hedge its bets' and maintain multiple hypotheses before viewing the available responses. Formally, we encode each of the k factoids per sample (we set $k = 4$ for all of our experiments following Zhang et al. (2021)). We use the output from the [CLS] token which is prepended to the factoids to obtain a fixed-size representation for each. We define this matrix as $\mathbf{H_z} \in \mathbb{R}^{k \times d} = \{\mathbf{z_1}, \mathbf{z_2}, ..., \mathbf{z_k}\}$. We perform dot product attention with these factoids, and add a skip-connection to obtain the joint representation $\mathbf{O_z}$:

$$\mathbf{O_z} = softmax(\mathbf{O_x}\mathbf{H_z^T})\mathbf{H_z} + \mathbf{O_x}$$

3.4 Response-selection module

We encode the response in the same way as the factoids to obtain a vector $\mathbf{y} \in \mathbb{R}^d$. Following (Humeau et al., 2020) we perform dot product attention using the response as a query and the joint representation as keys and values:

$$\mathbf{o_y} = softmax(\mathbf{y}\mathbf{O_z^T})\mathbf{O_z}$$

This is a form of pseudo-relevance feedback (PRF) (Cao et al., 2008), although unlike the original PRF which operated over sparse embeddings, this operates over dense embeddings. As each joint representation vector $\mathbf{o_z}$ may have attended to different information from the factoids, the model is able to select how to incorporate the factoids by first considering the available response candidates. This 'look before you leap' approach mitigates against the model predicting a response which has no corresponding candidate, which is a limitation of current retrieval models. We finally score the response by taking the dot product of each response embedding, with its corresponding PRF representation:

$$S(x, y) = \mathbf{o_y} \cdot \mathbf{y}$$

During training, following Henderson et al. (2017), we recycle the other response embeddings in the batch as negatives.

$$L_{NLL} = \sum_{i=1} S(x_i, y_i) - \sum_{i=1} \log \sum_{j=1} e^{S(x_i, y_j)}$$

3.5 Computing knowledge selection scores.

Unlike most knowledge-selection architectures, our model does not explicitly compute a probability distribution over documents $P(Z|x)$. Instead, we only have an interaction matrix $\in \mathbb{R}^{l \times k}$ between the context representation $\mathbf{O_x}$ and factoid embeddings $\mathbf{H_z}$. Empirically, we find that simply mean pooling over the context representation dimension obtains strong results [2]. For the purposes of evaluating the model with respect to knowledge-selection, we simply take the argmax of this vector as the chosen knowledge.

$$P(Z|x) = softmax(\frac{1}{L}\sum_{l=1}^{L} \mathbf{o_{x,l}}\mathbf{H_z^T})$$

Note however that this only provides a lower bound of the model's ability, as having multiple latent vectors allows the model to maintain competing hypotheses for knowledge-selection, creating more holistic interaction between contexts, factoids and responses.

4 Experiment

We conduct experiments on both knowledge-grounded and persona-grounded datasets (table 2), and report our results for both the knowledge-selection and the response-selection tasks. We compare our model both to our own baselines and results from other papers.

4.1 Datasets

Wizard of Wikipedia (WoW). (Dinan et al., 2019) Contains asymmetric dialogues between an 'apprentice' and a 'wizard', structured around a topic that both speakers are instructed to deep-dive. The wizard has access to extracts from Wikipedia (c.61 per turn) which they use to inject knowledge into the discussion. The test set is split into two subsets: test seen and test unseen. The former contains topics shared with the training data while the latter contains novel topics. The knowledge-selection task involves selecting the golden factoid from the c.61 candidates as chosen by the human wizard during dataset creation. We report both knowledge and response-selection.

Persona Chat (PC). (Zhang et al., 2018) Contains dialogues between crowdworkers, who are

[2] Max pooling also gave very similar results.

Dataset	Train	Valid	Test	# Turns	# Words
Wizard of Wikipedia	74,092	3,939	3,865	9.0	21.6
PersonaChat	131,438	7,801	6,634*	14.8	11.9

Table 2: Statistics for the datasets showing their sizes, average turns per dialogue and average words per utterance. *test data is not actually released, so we use the validation data where relevant instead.

instructed to get to know one another. Each of them is assigned a persona consisting of at least 5 short sentences. Similar to WoW, we treat each persona sentence as a latent factoid. The dataset also provides more challenging *revised* personas, which contain precisifications or generalisations of the *original* personas: e.g. 'I like playing sports' could become 'I play football every weekend' or vice versa. The response-selection task requires selecting the golden response from 19 other random candidates. We report response-selection only as we do not have knowledge-selection labels.

4.2 Baseline models

For strong comparison baselines, we select models that can pre-compute factoid representations and do not require ground truth knowledge-selection labels. We believe this is a more realistic setting given a) during inference the number of factoids is typically too large to recompute their representations dynamically, b) labelled <context, factoid> data is scarce and therefore not scalable.

Retrieval Augmented Retrieval. Similar to RAG models (Lewis et al., 2020) except in a bi-encoder retrieval setting: We marginalise over each <context, factoid> pair to obtain a single vector representation we compare against the response vectors. The generative version of this architecture has obtained state-of-the-art (SoTA) results on the WoW task (Shuster et al., 2021).

Concat Transformer (PC only). Bi-encoder which concatenates all persona sentences to dialogue context and encodes with all-on-all attention between context and persona. Although this approach is tangential to ours, as factoids cannot be pre-computed, we report it for completeness, given the current SoTA models use this approach (Ouguz et al., 2021; Wolf et al., 2019b).

Oracle Supervision (WoW only). Bi-encoder trained on pseudo-knowledge-labels selected by TF-IDF (Ramos, 2003) score with the response. We only evaluate on WoW where we have ground truth knowledge labels. This baseline is a scalable alternative to learning knowledge-selection end-to-end as we do.

Memory Network. One-hop memory network with a skip-connection, in which context acts as query and factoids as the memory vectors (Dinan et al., 2019). This baseline has shown strong results in persona-grounding (Mazaré et al., 2018), and contrasts against our model as it compresses the context and factoids into a single latent vector.

4.3 Implementation

We implement our models in PyTorch (Paszke et al., 2019) using HuggingFace's transformers library (Wolf et al., 2019a). We finetune our models from the TinyBERT (Jiao et al., 2020) checkpoint using the AdamW optimizer (Loshchilov and Hutter, 2017) with an initial learning-rate of $5e-5$ with linear decay, with a batch size of 128 for up to 10 epochs, until validation loss plateaus. We limit each factoid and response to 32 tokens, which is typically

Number of codes	Recall@1
$l = 1$	63.3
$l = 32$	68.4
$l = 128$	69.5
$l = 512$	70.1

Table 3: **Accuracy increases with number of codes** on the Persona Chat validation set with original persona.

Model	Seen	Unseen
MemNet	75.2	56.0
Retrieval Augmented Retrieval	73.8	55.4
PolyMemNet	78.2	58.3

Table 4: **Recall@1 on the response-selection task for WoW test sets.**

sufficient to avoid truncation. To enable multi-turn retrieval, for the context, we take the last 128 tokens which is capped at the last four turns for PC or last two turns for WoW [3]. To save memory, we only capture gradients for the top scoring factoids for each sample ($k = 4$). This prevents the embeddings becoming stale (Guu et al., 2020), while remaining memory-efficient.

4.4 Evaluation

We evaluate performance using *recall@1*, which measures the model's ability to select the golden factoid/response from a pool of candidates (Dinan et al., 2019). We use the validation data for PC instead of the test data which is not publicly available.

4.5 Results

End-to-end training beats noisy supervision. As shown in table 5, PolyMemNet significantly outperforms oracle supervision. This suggests knowledge-selection benefits from the additional loss signal of the response-selection task. The performance-gap is particularly noticeable on unseen topics, where our model often outperforms fully-supervised methods such as TMN and PostKS, suggesting supervision causes over-fitting. We believe this is an important result, as real users are unlikely to stick to the limited subset of topics covered in knowledge-grounded datasets. We underperform the supervised SKT model, however this has uses the BERT-base model which is sigificantly larger than TinyBERT which we use (110M parameters vs 14.5M).

Enriched representations from multiple latent codes. In table 5 we observe a more significant jump when grounding the Polyencoder with our method (+23.7%), compared with grounding a standard bi-encoder with MemNet (+11.5%) on PC with original persona. This suggests the performance gains are due to PolyMemNet making better use of knowledge, rather than merely the superiority of the Polyencoder over the bi-encoder. PolyMemNet achieves a 17% boost in knowledge-selection accuracy and 13% in response-selection accuracy versus MemNet. PolyMemNet is even comparable to the MemNet from Mazaré et al. (2018) which has extensive Reddit pretraining, showing our approach is also very sample efficient. Having multiple codes further allows PolyMemNet to model the one-to-many relation between contexts and knowledge and contexts and responses, as the architecture can maintain multiple 'hypothesis' vectors throughout the end-to-end process - this effect is demonstrated through increased performance as we increase the number of codes (table 3).

Late-stage interaction is a good approximator of full interaction. In tables 5 and 4, we find PolyMemNet outperforms more memory-intensive models which employ full token-level attention between contexts and factoids such as

[3] As shown in table 2, WoW utterances are on average twice as long as those in PC

Method	Memory usage	Persona Chat			Wiz. of Wikipedia	
		None	Original	Revised	Seen	Unseen
Starspace (Zhang et al., 2018)	–	31.8	49.1	32.2	–	–
Profile Memory (Zhang et al., 2018)	–	31.8	50.9	35.4	–	–
KV Profile Memory (Zhang et al., 2018)	–	**34.9**	**51.1**	35.1	–	–
MemNet (Mazaré et al., 2018)	–	–	–	42.1	–	–
MemNet † (Mazaré et al., 2018)	–	–	–	**60.7**	–	–
E2E TMN (w/ KL) (Dinan et al., 2019)	–	–	–	–	21.1	14.3
E2E TMN (no KL) (Dinan et al., 2019)	–	–	–	–	13.4	11.8
PostKS (w/ KL) (Lian et al., 2019)	–	–	–	–	23.4	9.4
PostKS (no KL) (Lian et al., 2019)	–	–	–	–	4.8	4.2
SKT+BERT (w/ KL) (Kim et al., 2020)	–	–	–	–	**26.8**	**18.3**
SKT+BERT (no KL) (Kim et al., 2020)	–	–	–	–	0.3	0.1
Random	–	5.0	5.0	5.0	2.7	2.3
TF-IDF	–	25.8	30.7	24.0	7.4	7.8
Bi-encoder	–	55.0	–	–	–	–
Polyencoder ($l = 128$)	–	**56.2**	–	–	–	–
Oracle supervision	–	–	–	–	14.7	14.0
Concat Transformer (CT)	2.22x	–	42.7	45.4	–	–
MemNet	1.00x	–	61.3	57.0	13.4	13.2
Retrieval Augmented Retrieval (RAR)	4.20x	–	65.5	57.6	9.8	9.2
PolyMemNet ($l = 128$)	1.03x	–	**69.5****	**60.3****	**15.7***	**14.7**

Table 5: **Recall@1 for response-selection on the Persona Chat validation set and knowledge-selection on the Wizard of Wikipedia test set.** Memory usage is based on average usage during training on the Persona Chat w/ original persona task. KL (= knowledge loss) indicates whether the model was additionally supervised on the golden knowledge labels. † = with pretraining on 1.7B Reddit comments. Statistical significance tests were conducted between PolyMemNet and the next best model, where * and ** denote $p < 0.05$ and $p < 0.01$ respectively.

CT and RAR, despite being up to 4x more memory efficient. We attribute the outperformance to a combination of the de-noising effect of vector-based methods, given factoids are known to be noisy at the token-level (Zheng et al., 2021), as well as the ability to condition a prediction on multiple factoids (compared to being treated separately in RAR).

5 Conclusion

In this work we have presented a new architecture for the task of unsupervised knowledge-selection and grounded response-selection in an end-to-end setting. Our PolyMemNet model extends previous late-stage interaction retrieval frameworks to the grounded dialogue setting, allowing for richer context-factoid-response interaction, without materially increasing memory footprint. In knowledge-selection, particularly on unseen topics, it is even able to close the gap and in some cases outperform models supervised on knowledge labels.

Future work might consider how to extend the interaction between latent codes and factoids, such as by adding multiple hops to the interaction, or learning weights for the attention process. Additionally, the latent state might benefit from additional self-attention and feed-forward layers, as in a transformer.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 2020. Bart for knowledge grounded conversations. In *Converse@KDD*.

Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen E. Robertson. 2008. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR '08*.

Danqi Chen, Adam Fisch, J. Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *ACL*.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and J. Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. *ArXiv*, abs/1811.01241.

Angela Fan, Claire Gardent, C. Braud, and Antoine Bordes. 2021. Augmenting transformers with knn-based composite memory for dialog. *Transactions of the Association for Computational Linguistics*, 9:82–99.

Kelvin Guu, Kenton Lee, Z. Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *ICML*.

Matthew Henderson, Rami Al-Rfou, B. Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and R. Kurzweil. 2017. Efficient natural language response suggestion for smart reply. *ArXiv*, abs/1705.00652.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and J. Weston. 2020. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *ICLR*.

Eric Jang, Shixiang Shane Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. *ArXiv*, abs/1611.01144.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling bert for natural language understanding. *ArXiv*, abs/1909.10351.

O. Khattab, Christopher Potts, and Matei A. Zaharia. 2021. Relevance-guided supervision for openqa with colbert. *Transactions of the Association for Computational Linguistics*, 9:929–944.

Byeongchang Kim, Jaewoo Ahn, and Gunhee Kim. 2020. Sequential latent knowledge selection for knowledge-grounded dialogue. *ArXiv*, abs/2002.07510.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, M. Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401.

Jiwei Li, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, and W. Dolan. 2016. A persona-based neural conversation model. *ArXiv*, abs/1603.06155.

Rongzhong Lian, Min Xie, Fan Wang, Jinhua Peng, and Hua Wu. 2019. Learning to select knowledge for response generation in dialog systems. *ArXiv*, abs/1902.04911.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101.

Pierre-Emmanuel Mazaré, Samuel Humeau, Martin Raison, and Antoine Bordes. 2018. Training millions of personalized dialogue agents. In *EMNLP*.

Nikita Moghe, Siddharth Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. *ArXiv*, abs/1809.08205.

N. Mostafazadeh, Chris Brockett, W. Dolan, Michel Galley, Jianfeng Gao, Georgios P. Spithourakis, and Lucy Vanderwende. 2017. Image-grounded conversations: Multimodal context for natural question and response generation. In *IJCNLP*.

Barlas Oguz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Wen tau Yih, Sonal Gupta, and Yashar Mehdad. 2021. Domain-matched pre-training tasks for dense retrieval. *ArXiv*, abs/2107.13602.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Juan Enrique Ramos. 2003. Using tf-idf to determine word relevance in document queries.

S. Robertson and H. Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3:333–389.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and J. Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *ArXiv*, abs/2104.07567.

Sainbayar Sukhbaatar, Arthur D. Szlam, J. Weston, and R. Fergus. 2015. End-to-end memory networks. In *NIPS*.

Elena Voita, David Talbot, F. Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *ACL*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019a. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *ArXiv*, abs/1901.08149.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur D. Szlam, Douwe Kiela, and J. Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In *ACL*.

Yizhe Zhang, Siqi Sun, Xiang Gao, Yuwei Fang, Chris Brockett, Michel Galley, Jianfeng Gao, and Bill Dolan. 2021. Joint retrieval and generation training for grounded text generation. *ArXiv*, abs/2105.06597.

Wen Zheng, Natasa Milic-Frayling, and Ke Zhou. 2021. Knowledge-grounded dialogue generation with term-level de-noising. In *FINDINGS*.