# Training Neural Response Selection for Task-Oriented Dialogue Systems

**Matthew Henderson, Ivan Vulić, Daniela Gerz, Iñigo Casanueva, Paweł Budzianowski, Sam Coope, Georgios Spithourakis, Tsung-Hsien Wen, Nikola Mrkšić,** and **Pei-Hao Su**

PolyAI Limited
London, United Kingdom

matt@poly-ai.com ivan@poly-ai.com



## Abstract

Despite their popularity in the chatbot literature, retrieval-based models have had modest impact on task-oriented dialogue systems, with the main obstacle to their application being the low-data regime of most task-oriented dialogue tasks. Inspired by the recent success of pretraining in language modelling, we propose an effective method for deploying *response selection* in task-oriented dialogue. To train response selection models for task-oriented dialogue tasks, we propose a novel method which: **1)** pretrains the response selection model on large general-domain conversational corpora; and then **2)** fine-tunes the pretrained model for the target dialogue domain, relying only on the small in-domain dataset to capture the nuances of the given dialogue domain. Our evaluation on six diverse application domains, ranging from e-commerce to banking, demonstrates the effectiveness of the proposed training method.

## 1 Introduction

Retrieval-based dialogue systems conduct conversations by selecting the most appropriate system response given the dialogue history and the input user utterance (i.e., the full dialogue context). A typical retrieval-based approach to dialogue encodes the input and a large set of responses in a joint semantic space. When framed as an ad-hoc retrieval task (Deerwester et al., 1990; Ji et al., 2014; Kannan et al., 2016; Henderson et al., 2017), the system treats each input utterance as a *query* and retrieves the most relevant response from a large response collection by computing semantic similarity between the query representation and the encoding of each response in the collection. This task is referred to as *response selection* (Wang et al., 2013; Al-Rfou et al., 2016; Yang et al., 2018; Du and Black, 2018; Chaudhuri et al., 2018; Weston et al., 2018), as illustrated in Figure 1.

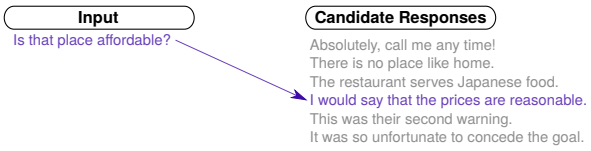



Figure 1: The conversational response selection task: given the input sentence, the goal is to identify the relevant response from a large collection of candidates.

Formulating dialogue as a response selection task stands in contrast with other data-driven dialogue modeling paradigms such as modular and end-to-end task-based dialogue systems (Young, 2010; Wen et al., 2017b; Liu and Perez, 2017; Li et al., 2017; Bordes et al., 2017). Unlike standard task-based systems, response selection does not rely on explicit task-tailored semantics in the form of domain ontologies, which are hand-crafted for each task by domain experts (Henderson et al., 2014a,b; Mrkšić et al., 2015). Response selection also differs from chatbot-style systems which generate new responses by generalising over training data, their main deficiency being the tendency towards generating universal but irrelevant responses such as *"I don't know"* or *"Thanks"* (Vinyals and Le, 2015; Li et al., 2016; Serban et al., 2016; Song et al., 2018). Therefore, response selection removes the need to engineer structured domain ontologies, and to solve the difficult task of general language generation. Furthermore, it is also much easier to constrain or combine the output of response selection models. This design also bypasses the construction of dedicated decision-making policy modules.

Although conceptually attractive, retrieval-based dialogue systems still suffer from *data scarcity*, as deployment to a new domain requires a sufficiently large in-domain dataset for training the response selection model. Procuring such data is expensive and labour-intensive, with annotated datasets for task-based dialogue still few and far between, as

well as limited in size.[1]

Recent work on language modelling (LM) pretraining (Peters et al., 2018; Howard and Ruder, 2018) has shown that task-specific architectures are not necessary in a number of NLP tasks. The best results have been achieved by LM pretraining on large unannotated corpora, followed by supervised fine-tuning on the task at hand (Devlin et al., 2019). Given the compelling benefits of large-scale pretraining, our work poses a revamped question for response selection: can we *pretrain* a general response selection model and then *adapt* it to a variety of different dialogue domains?

To tackle this problem, we propose a two-step training procedure which: **1)** pretrains a response selection model on large conversational corpora (such as Reddit); and then **2)** fine-tunes the pretrained model for the target dialogue domain. Throughout the evaluation, we aim to provide answers to the following two questions:

1. **(Q1)** *How to pretrain?* Which encoder structure can best model the Reddit data?

2. **(Q2)** *How to fine-tune?* Which method can efficiently adapt the pretrained model to a spectrum of target dialogue domains?

Regarding the first question, the results support findings from prior work (Cer et al., 2018; Yang et al., 2018): the best scores are reported with simple transformer-style architectures (Vaswani et al., 2017) for input-response encodings. Most importantly, our results suggest that pretraining plus fine-tuning for response selection is useful across six different target domains.

As for the second question, the most effective training schemes are lightweight: the model is pretrained only once on the large Reddit training corpus, and the target task adaptation does not require expensive retraining on Reddit. We also show that the proposed two-step response selection training regime is more effective than directly applying off-the-shelf state-of-the-art sentence encoders (Cer et al., 2018; Devlin et al., 2019).

[1]For instance, the recently published MultiWOZ dataset (Budzianowski et al., 2018) comprises a total of 115,424 dialogue turns scattered over 7 target domains. It is several times larger than other standard task-based dialogue datasets such as DSTC2 (Henderson et al., 2014b) with 23,354 turns, Frames (El Asri et al., 2017) with 19,986 turns, or M2M (Shah et al., 2018) with 14,796 turns. To illustrate the difference in magnitude, the Reddit corpus used in this work for response selection pretraining comprises 727M dialogue turns.

We hope that this paper will inform future development of response-based task-oriented dialogue. Training and test datasets, described in more detail by Henderson et al. (2019), are available at: `github.com/PolyAI-LDN/conversational-datasets`.

## 2 Methodology

**Why Pretrain and Fine-Tune?** By simplifying the conversational learning task to a response selection task, we can relate target domain tasks to general-domain conversational data such as Reddit (Al-Rfou et al., 2016). This also means that parameters of response selection models in target domains with scarce training resources can be initialised by a general-domain pretrained model.

The proposed two-step approach, described in §2.1 and §2.2, can be seen as a "lightweight" task adaptation strategy: the expensive Reddit model pretraining is run only once (i.e., training time is typically measured *in days*), and the model is then fine-tuned on $N$ target tasks (i.e., fine-tuning time is *in minutes*). The alternatives are "heavyweight" data mixing strategies. First, in-domain and Reddit data can be fused into a single training set: besides expensive retraining for each task, the disbalance between in-domain and Reddit data sizes effectively erases the target task signal. An improved data mixing strategy keeps the identities of the origin datasets (Reddit vs. target) as features in training. While this now retains the target signal, our preliminary experiments indicated that the results again fall short of the proposed lightweight fine-tuning strategies. In addition, this strategy still relies on expensive Reddit retraining for each task.

### 2.1 Step 1: Response Selection Pretraining

**Reddit Data.** Our pretraining method is based on the large Reddit dataset compiled and made publicly available recently by Henderson et al. (2019). This dataset is suitable for response selection pretraining due to multiple reasons as discussed by Al-Rfou et al. (2016). First, the dataset offers organic conversational structure and it is large at the same time: all Reddit data from January 2015 to December 2018, available as a BigQuery dataset, span almost 3.7B comments. After preprocessing the dataset to remove both uninformative and long comments[2] and pairing all comments with their

[2]We retain only sentences containing more than 8 and less than 128 word tokens.

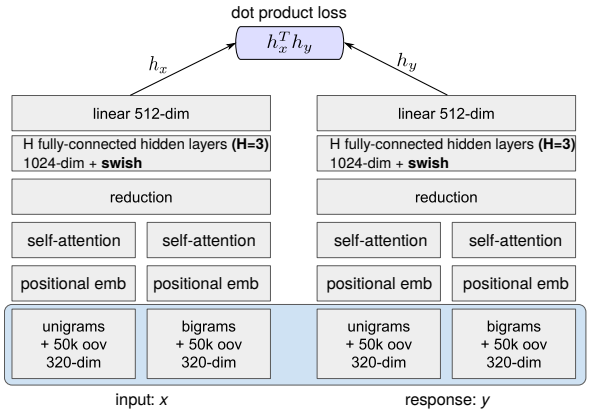



Figure 2: Schematic input-response encoder model structure. We show the best-performing architecture for brevity, while we evaluate a variety of other encoder architecture configurations later in §4.1.

responses, we obtain more than 727M comment-response pairs which are used for model pretraining. This Reddit dataset is substantially larger than the previous Reddit dataset of Al-Rfou et al. (2016), which spans around 2.1B comments and 133M conversational threads, and is not publicly available. Second, Reddit is extremely diverse topically (Schrading et al., 2015; Al-Rfou et al., 2016): there are more than 300,000 sub-forums (i.e., subreddits) covering diverse topics of discussion. Finally, compared to message-length-restricted Twitter conversations (Ritter et al., 2010), Reddit conversations tend to be more natural. In summary, all these favourable properties hold promise to support a large spectrum of diverse conversational domains.

**Input and Response Representation.** We now turn to describing the architecture of the main pretraining model. The actual description focuses on the best-performing architecture shown in Figure 2, but we also provide a comparative analysis of other architectural choices later in §4.1.

First, similar to Henderson et al. (2017), raw text is converted to unigrams and bigrams, that is, we extract $n$-gram features from each input $x$ and its corresponding response $y$ from (Reddit) training data. During training we obtain $d$-dimensional feature representations ($d = 320$, see Figure 2) shared between inputs and responses for each unigram and bigram jointly with other neural net parameters. In addition, the model can deal with out-of-vocabulary unigrams and bigrams by assigning a random id from 0 to 50,000 to each, which is then used to look up their embedding. When fine-tuning, this allows the model to learn representations of words that otherwise would be out-of-vocabulary.

**Sentence Encoders.** The unigram and bigram embeddings then undergo a series of transformations on both the input and the response side, see Figure 2 again. Following the transformer architecture (Vaswani et al., 2017), positional embeddings and self-attention are applied to unigrams and bigrams separately. The representations are then combined as follows (i.e., this refers to the *reduction* layer in Figure 2): the unigram and bigram embeddings are each summed and divided by the square root of the word sequence length. The two vectors are then averaged to give a single 320-dimensional representation of the text (input or response).

The averaged vector is then passed through a series of $H$ fully connected $h$-dim feed-forward hidden layers ($H = 3$; $h = 1,024$) with *swish* as the non-linear activation, defined as: $swish(x) = x \cdot \text{sigmoid}(\beta x)$ (Ramachandran et al., 2017).[3] The final layer is linear and maps the text into the final $l$-dimensional ($l = 512$) representation: $h_x$ for the input text, and $h_y$ for the accompanying response text. This provides a fast encoding of the text, with some sequential information preserved.[4]

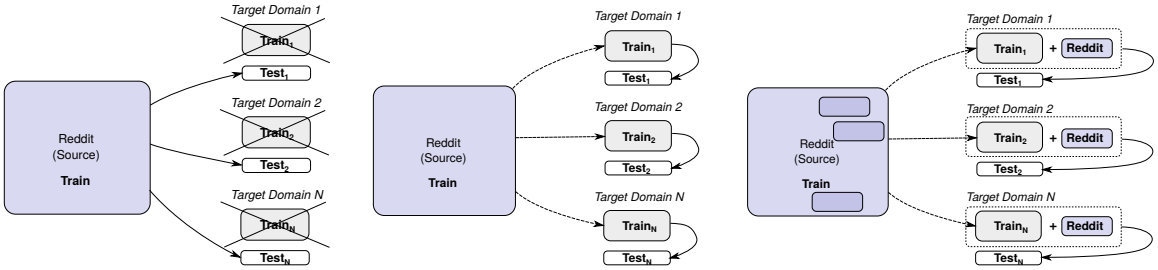
**Scaled Cosine Similarity Scoring.** The relevance of each response to the given input is then quantified by the score $S(x, y)$. It is computed as scaled cosine similarity: $S(x, y) = C \cdot cos(h_x, h_y)$, where $C$ is a learned constant, constrained to lie between 0 and $\sqrt{l}$. We resort to scaled cosine similarity instead of general dot product as the absolute values are meaningful for the former. In consequence, the scores can be thresholded, and retrained models can rely on the same thresholding.

Training proceeds in batches of $K$ *(input, response)* pairs $(x_1, y_1), \ldots, (x_K, y_K)$. The objective tries to distinguish between the true relevant response and irrelevant/random responses for each input sentence $x_i$. The training objective for a single batch of $K$ pairs is as follows:

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

[3] We fix $\beta = 1$ as suggested by Ramachandran et al. (2017). The use of *swish* is strictly empirically driven: it yielded slightly better results in our preliminary experiments than the alternatives such as *tanh* or a family of LU/ReLU-related activations (He et al., 2015; Klambauer et al., 2017).

[4] Experiments with higher-order n-grams, recurrent, and convolutional structures have not provided any substantial gain, and slow down the encoder model considerably.




(a) REDDIT-DIRECT (b) FT-DIRECT (c) FT-MIXED

Figure 3: High-level overview of baseline and fine-tuning strategies used in our evaluation. **(a)** REDDIT-DIRECT: a pretrained general-domain (Reddit) response selection model is directly applied on each target task, without any target domain fine-tuning; **(b)** FT-DIRECT: after pretraining the large response selection model on Reddit, the model is fine-tuned for each target task by directly continuing the training on (much smaller) target domain data; **(c)** FT-MIXED: similar to FT-DIRECT, but the crucial difference is in-batch mixing of Reddit input-response pairs with target domain pairs during the target fine-tuning procedure. Another baseline (TARGET-ONLY) trains a response selection model on each target task separately without leveraging general-domain Reddit data (not shown).

Effectively, Eq. (1) maximises the score of pairs $(x_i, y_i)$ that go together in training, while minimising the score of pairing each input $x_i$ with $K'$ negative examples, that is, responses that are not associated with the input $x_i$. For simplicity, as in prior work (Henderson et al., 2017; Yang et al., 2018), for each input $x_i$, we treat all other $K-1$ responses in the current batch $y_j \neq y_i$ as negative examples.[5]

As discussed by Henderson et al. (2017) in the context of e-mail reply applications, this design enables efficient response search as it allows for precomputing vectors of candidate responses independently of input queries, and searching for responses with high scaled cosine similarity scores in the precomputed set. It also allows for approximate nearest neighbour search (Malkov and Yashunin, 2016) which speeds up computations drastically at the modest decrease of retrieval performance.[6]

Finally, in this work we rely on a simple strategy based on random negative examples. In future work, we plan to experiment with alternative (non-random) negative sampling strategies. For instance, inspired by prior work on semantic specialisation (Mrkšić et al., 2017b) and parallel corpora mining (Guo et al., 2018), difficult negative examples might comprise invalid responses that are semantically related to the correct response (measured by e.g. dot-product similarity).

[5]Note that the matrix $\mathbf{S} = C \cdot [h_{y_1}, \ldots, h_{y_K}] \cdot [h_{x_1}, \ldots, h_{x_K}]^T$ is inexpensive to compute.

[6]E.g., experiments on Reddit test data reveal a 130× speed-up using the approximate search method of Malkov and Yashunin (2016) while retaining 95% top-30 recall.

## 2.2 Step 2: Target Domain Fine-Tuning

The second step concerns the application of the pretrained general Reddit model on $N$ target domains. We assume that we have the respective training and test sets of $K_{N,tr}$ and $K_{N,te}$ in-domain input-response pairs for each of the $N$ domains, where $K_{N,tr}$ and $K_{N,te}$ are considerably smaller than the number of Reddit training pairs. We test two general fine-tuning strategies, illustrated in Figure 3.

FT-DIRECT directly continues where the Reddit pretraining stopped: it fine-tunes the model parameters by feeding the $K_{N,tr}$ in-domain *(input, response)* pairs into the model and by following exactly the same training principle as described in §2.1. The fine-tuned model is then tested in the in-domain response selection task using $K_{N,te}$ test pairs, see Figure 3b.

FT-MIXED attempts to prevent the "specialisation" of the Reddit model to a single target domain, that is, it aims to maintain stable performance on the general-domain Reddit data. This way, the model can support multiple target tasks simultaneously. Instead of relying only on in-domain training pairs, we now perform in-batch mixing of Reddit pairs with in-domain pairs: $M\%$ of the pairs in each batch during fine-tuning are Reddit pairs, while $(100-M)\%$ of the pairs are in-domain pairs, where $M$ is a tunable hyper-parameter. With this fine-tuning strategy, outlined in Figure 3c, each dataset provides negative examples for the other one, enriching the learning signal.

We compare FT-DIRECT and FT-MIXED against two straightforward and insightful baselines: the REDDIT-DIRECT model from Figure 3a directly ap-

plies the pretrained Reddit model on the target task without any in-domain fine-tuning. Comparisons to this baseline reveal the importance of fine-tuning. On the other hand, the TARGET-ONLY baseline simply trains the response selection model from Figure 2 from scratch directly on the in-domain $K_{N,tr}$ pairs. Comparisons to this baseline reveal the importance of Reddit pretraining. For all TARGET-ONLY models in all target tasks, we tuned the word embedding sizes and embedding dropout rates on the corresponding training sets.

## 3 Experimental Setup

**Training Setup and Hyper-Parameters.** All input text is lower-cased and tokenised, numbers with 5 or more digits get their digits replaced by a wildcard symbol #, while words longer than 16 characters are replaced by a wildcard token LONGWORD. Sentence boundary tokens <*S*> and </*S*> are added to each sentence. The vocabulary consists of the unigrams that occur at least 10 times in a random 1M subset of the Reddit training set –this results in a total of 105K unigrams– plus the 200K most frequent bigrams in the same random subset.

The following training setup refers to the final Reddit model, illustrated in Figure 2, and used in fine-tuning. The model is trained by SGD setting the initial learning rate to 0.03, and then decaying the learning rate by 0.3x every 1M training steps after the first 2.5M steps. Similar to learning rate scaling by the batch size used in prior work (Goyal et al., 2017; Codreanu et al., 2017), we scale the unigram and bigram embedding gradients by the batch size. The batch size is 500, and attention projection dimensionality is 64.

We also apply the label smoothing technique (Szegedy et al., 2016), shown to reduce overfitting by preventing a network to assign full probability to the correct training example (Pereyra et al., 2017). Effectively, this reshapes Eq. (1): each positive training example in each batch gets assigned the probability of 0.8, while the remaining probability mass gets evenly redistributed across in-batch negative examples. Finally, we train the model on 13 GPU nodes with one Tesla K80 each for 18 hours: the model sees around 2B examples and it is sufficient for the model to reach convergence.[7] Fine-tuning is run by relying on early stopping on in-domain validation data. The ratio of Reddit and in-domain pairs with FT-MIXED is set to 3:1 (in favour of Reddit) in all experimental runs.

**Test Domains and Datasets.** We conduct experiments on six target domains with different properties and varying corpora sizes. The diversity of evaluation probes the robustness of the proposed pretraining and fine-tuning regime. The summary of target domains and the corresponding data is provided in Table 1. All datasets are in the form of *(input, response)* pairs. For UBUNTU[8], SEMEVAL15[9], and AMAZONQA[10] we use standard data splits into training, dev, and test portions following the original work (Lowe et al., 2017; Nakov et al., 2015; Wan and McAuley, 2016). For the OpenSubtitles dataset (OPENSUB) (Lison and Tiedemann, 2016), we rely on the data splits introduced by Henderson et al. (2019). We evaluate pretrained Reddit models on the REDDIT held-out data: 50K randomly sampled *(input, response)* pairs are used for testing.

We have also created a new FAQ-style dataset in the e-banking domain which includes question-answer pairs divided into 77 unique categories with well-defined semantics (e.g., "card activation", "closing account", "refund request"). Such FAQ information can be found in various e-banking customer support pages, but the answers are highly hierarchical and often difficult to locate. Our goal is to test the fine-tuned encoder's ability to select the relevant answers to the posed question. To this end, for each question we have collected 10 paraphrases that map to the same answer. All unique *(question, answer)* pairs are added to the final dataset, which is then divided into training (70%), validation (20%) and test portions (10%), see Table 1.

**Baseline Models.** Besides the direct encoder model training on each target domain without pretraining (TARGET-ONLY), we also evaluate two standard IR baselines based on keyword matching: 1) a simple TF-IDF query-response scoring (Manning et al., 2008), and 2) Okapi BM25 (Robertson and Zaragoza, 2009).

Furthermore, we also analyse how pretraining plus fine-tuning for response selection compares to a representative sample of publicly available neural network embedding models which embed inputs and responses into a vector space. We include the following embedding models, all of

[7] Training is relatively cheap compared to other large models: e.g., BERT models (Devlin et al., 2019) were pre-trained for 4 days using 4 Cloud TPUs (BERT-SMALL) or 16 Cloud TPUs (BERT-LARGE).

[8] https://github.com/rkadlec/

[9] http://alt.qcri.org/semeval2015/task3/

[10] http://jmcauley.ucsd.edu/data/amazon/qa/

| Dataset | Reference | Domain | Training Size | Test Size |
|---|---|---|---|---|
| REDDIT | (Henderson et al., 2019) | discussions on various topics | 654,396,778 | 72,616,937 |
| OPENSUB | (Lison and Tiedemann, 2016) | movies, TV shows | 283,651,561 | 33,240,156 |
| AMAZONQA | (Wan and McAuley, 2016) | e-commerce, retail | 3,316,905 | 373,007 |
| UBUNTU | (Lowe et al., 2017) | computers, technical chats | 3,954,134 | 72,763 |
| BANKING | New | e-banking applications, banking FAQ | 10,395 | 1,485 |
| SEMEVAL15 | (Nakov et al., 2015) | lifestyle, tourist and residential info | 9,680 | 1,158 |

Table 1: Summary of all target domains and data. Data sizes: a total number of unique *(input, response)* pairs. Note that some datasets contain many-to-one pairings (i.e., multiple inputs are followed by the same response; BANKING) and one-to-many pairings (i.e., one input generates more than one plausible response; SEMEVAL15).

which are readily available online.[11] (1) Universal Sentence Encoder of Cer et al. (2018) is trained using a transformer-style architecture (Vaswani et al., 2017) on a variety of web sources such as Wikipedia, web news, discussion forums as well as on the Reddit data. We experiment with the base USE model and its larger variant (USE-LARGE). (2) We run fixed mean-pooling of ELMO contextualised embeddings (Peters et al., 2018) pretrained on the bidirectional LM task using the LM 1B words benchmark (Chelba et al., 2013): ELMO. (3) We also compare to two variants of the bidirectional transformer model of Devlin et al. (2019) (BERT-SMALL and BERT-LARGE).[12]

We compare to two model variants for each of the above vector-based baseline models. First, the SIM method ranks responses according to their cosine similarity with the context vector: it relies solely on pretrained models without any further fine-tuning or adaptation, that is, it does not use the training set at all. The MAP variant learns a linear mapping on top of the response vector. The final score of a response with vector $h_y$ for an input with vector $h_x$ is the cosine similarity $cos(\cdot, \cdot)$ of the context vector with the mapped response vector:

$$cos\big(h_x, (W + \alpha I) \cdot h_y\big). \quad (2)$$

$W$, $\alpha$ are parameters learned on a random sample of 10,000 examples from the training set using the same dot product loss from Eq. (1), and $I$ is the identity matrix. Vectors are $\ell_2$-normalised before being fed to the MAP method. For all baseline models, learning rate and regularization parameters are tuned using a held-out development set.

[11] https://www.tensorflow.org/hub

[12] Note that the encoder architectures similar to the ones used by USE can also be used in the Reddit pretraining phase in lieu of the architecture shown in Figure 2. However, the main goal is to establish the importance of target response selection fine-tuning by comparing it to direct application of state-of-the-art pretrained encoders, used to encode both input and responses in the target domain.

| **Full Reddit Model** | **61.3** |
|---|---|
| - Wider hidden layers; $h = 2,048$, 24h training | 61.1 |
| - Narrower hidden layers; $h = 750$, 18h training | 60.8 |
| - Narrower hidden layers; $h = 512$ | 59.8 |
| - Batch size 50 (before 500) | 57.4 |
| - $H = 2$ (before $H = 3$) | 56.9 |
| - *tanh* activation (before *swish*) | 56.1 |
| - no label smoothing | 55.3 |
| - no self-attention | 48.7 |
| - remove bigrams | 35.5 |

Table 2: The results of different encoder configurations on the Reddit test data ($R_{100}$@1 scores $\times 100\%$). Starting from the full model (top row), each subsequent row shows a configuration with one component removed or edited from the configuration from the previous row.

The combination of the two model variants with the vector-based models results in a total of 10 baseline methods, as listed in Table 3.

**Evaluation Protocol.** We rely on a standard IR evaluation measure used in prior work on retrieval-based dialogue (Lowe et al., 2017; Zhou et al., 2018; Chaudhuri et al., 2018): *Recall@k*. Given a set of $N$ responses to the given input/query, where only one response is relevant, it indicates whether the relevant response occurs in the top $k$ ranked candidate responses. We refer to this evaluation measure as $\mathbf{R}_N@k$, and set $N = 100; k = 1$: $\mathbf{R}_{100}@1$. This effectively means that for each query, we indicate if the correct response is the top ranked response between 100 candidates. The final score is the average across all queries.

## 4 Results and Discussion

This section aims to provide answers to the two main questions posed in §1: which encoder architectures are more suitable for pretraining (Q1; §4.1), and how to adapt/fine-tune the pretrained model to target tasks (Q2; §4.2).

### 4.1 Reddit Pretraining

The full encoder model is described in §2.1 and visualised in Figure 2. In what follows, we also anal-

| | REDDIT | OPENSUB | AMAZONQA | UBUNTU | BANKING | SEMEVAL15 |
|---|---|---|---|---|---|---|
| TF-IDF | 26.7 | 10.9 | 51.8 | 27.5 | 27.3 | 38.0 |
| BM25 | 27.6 | 10.9 | 52.3 | 19.6 | 23.4 | 35.5 |
| USE-SIM | 36.6 | 13.6 | 47.6 | 11.5 | 18.2 | 36.0 |
| USE-MAP | 40.8 | 15.8 | 54.4 | 17.2 | 79.2 | 45.5 |
| USE-LARGE-SIM | 41.4 | 14.9 | 51.3 | 13.6 | 27.3 | 44.0 |
| USE-LARGE-MAP | 47.7 | 18.0 | 61.9 | 18.5 | 81.8 | **56.5** |
| ELMO-SIM | 12.5 | 9.5 | 16.0 | 3.5 | 6.5 | 19.5 |
| ELMO-MAP | 19.3 | 12.3 | 33.0 | 6.2 | 87.0 | 34.5 |
| BERT-SMALL-SIM | 17.1 | 13.8 | 27.8 | 4.1 | 13.0 | 13.0 |
| BERT-SMALL-MAP | 24.5 | 17.5 | 45.8 | 9.0 | 77.9 | 37.5 |
| BERT-LARGE-SIM | 14.8 | 12.2 | 25.9 | 3.6 | 10.4 | 10.0 |
| BERT-LARGE-MAP | 24.0 | 16.8 | 44.1 | 8.0 | 68.8 | 34.5 |
| REDDIT-DIRECT | **61.3** | 19.1 | 61.4 | 9.6 | 27.3 | 46.0 |
| TARGET-ONLY | - | 29.0 (18.2) | 83.3 (11.6) | 6.2 ( 2.3) | 88.3 ( 1.2) | 7.5 ( 1.1) |
| FT-DIRECT | - | **30.6** (40.0) | **84.2** (30.8) | **38.7** (51.9) | **94.8** (55.3) | 52.5 (55.2) |
| FT-MIXED | - | 25.5 (60.0) | 77.0 (59.6) | 38.1 (59.4) | 90.9 (59.8) | **56.5** (59.4) |

Table 3: Summary of the results ($R_{100}$@1 scores $\times 100\%$) with fine-tuning on all six target domains. Datasets are ordered left to right based on their size. The scores in the parentheses in the TARGET-ONLY, FT-DIRECT and FT-MIXED rows give the performance on the general-domain REDDIT test data. The scores are computed with de-duplicated inputs for SEMEVAL15 (i.e., the initial dataset links more responses to the same input), and de-duplicated answers for banking.

yse performance of other encoder configurations, which can be seen as ablated or varied versions of the full model. The results on the REDDIT response selection task are summarised in Table 2.

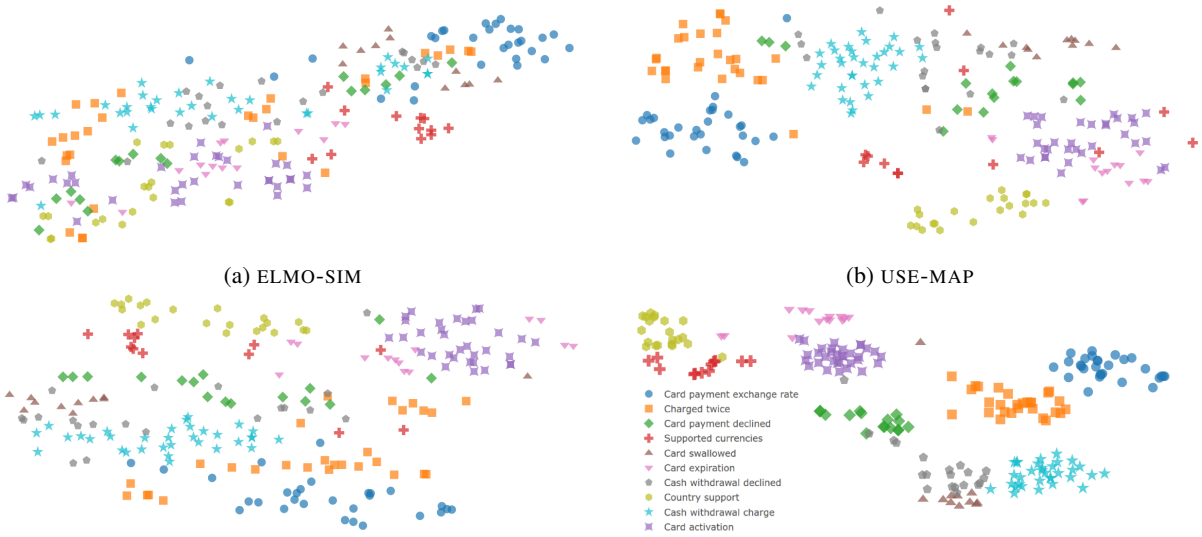
**Results and Discussion.** The scores suggest that the final model gets contribution from its multiple components: e.g., replacing *tanh* with the recently proposed *swish* activation (Ramachandran et al., 2017) is useful, and label smoothing also helps. Despite contradictory findings from prior work related to the batch size (e.g., compare (Smith et al., 2017) and (Masters and Luschi, 2018)), we obtain better results with larger batches. This is intuitive given the model design: increasing the batch size in fact means learning from a larger number of negative examples. The results also suggest that the model saturates when provided with a sufficient number of parameters, as wider hidden layers and longer training times did not yield any substantial gains.

The scores also show the benefits of self-attention and positional embeddings instead of deep feed-forward averaging of the input unigram and bigram embeddings (Iyyer et al., 2015). This is in line with prior work on sentence encoders (Cer et al., 2018; Yang et al., 2018), which reports similar gains on several classification tasks. Finally, we observe a large gap with the unigram-only model variant, confirming the importance of implicitly representing underlying sequences with $n$-grams (Henderson et al., 2017; Mrkšić et al., 2017a). Following the results, we fix the pretraining model in all follow-up experiments (top row in Table 2).

### 4.2 Target-Domain Fine-Tuning

**Results and Discussion.** The main results on all target tasks after fine-tuning are summarised in Table 3. First, the benefits of Reddit pretraining and fine-tuning are observed in all tasks regardless of the in-domain data size. We report large gains over the TARGET-ONLY model (which trains a domain-specific response selection encoder from scratch) especially for tasks with smaller training datasets (e.g., BANKING, SEMEVAL15). The low scores of TARGET-ONLY with smaller training data suggest overfitting: the encoder architecture cannot see enough training examples to learn to generalise. The gains are also present even when TARGET-ONLY gets to see much more in-domain input-response training data: e.g., we see slight improvements on OPENSUB and AMAZONQA, and large gains on UBUNTU when relying on the FT-DIRECT fine-tuning variant.

What is more, a comparison to REDDIT-DIRECT further suggests that fine-tuning even with a small amount of in-domain data can lead to large improvements: e.g., the gains over REDDIT-DIRECT are +67.5% on BANKING, +32.5% on UBUNTU, +22.8% on AMAZONQA, and +11.5% on OPENSUB. These results lead to the following crucial conclusion: while in-domain data are insufficient to train response selection models from scratch for many target domains, such data are invaluable for adapting a pretrained general-domain model to the target domain. In other words, the results indicate that the synergy between the abundant response




(a) ELMO-SIM (b) USE-MAP

(c) REDDIT-DIRECT (no fine-tuning) (d) FT-MIXED (with fine-tuning)

Figure 4: t-SNE plots (van der Maaten and Hinton, 2012) of encoded questions/inputs for a selection of 10 categories from the BANKING test set. The most coherent clusters for each category with well-defined semantics are observed with the FT-MIXED fine-tuning model applied on top of Reddit response selection pretraining.

selection Reddit data and scarce in-domain data is effectively achieved through the proposed training regime, and both components are crucial for the final improved performance in each target domain. In simple words, this finding confirms the importance of fine-tuning for the response selection task.

**Comparison to Baselines.** The results of TF-IDF and BM25 reveal that lexical evidence from the preceding input can partially help in the response selection task and it achieves reasonable performance across the target tasks. For instance, on some tasks (e.g., AMAZONQA, BANKING), such keyword matching baselines even outperform some of the vector-based baseline models, and are comparable to the REDDIT-DIRECT model variant. They are particularly strong for AMAZONQA and UBUNTU, possibly because rare and technical words (e.g., the product name) are very informative in these domains. However, these baselines are substantially outperformed by the proposed fine-tuning approach across the board.

A comparison to other pretrained sentence encoders in Table 3 further stresses the importance of training for the response selection task in particular. Using off-the-shelf sentence encoders such as USE or BERT directly on in-domain sentences without distinguishing the input and the response space leads to degraded performance compared even to TF-IDF, or the REDDIT-DIRECT baseline without in-domain fine-tuning. The importance of learning the mapping from input to response versus simply relying on similarity is also exemplified by the comparison between the MAP method and the simple SIM method: regardless of the actual absolute performance, MAP leads to substantial gains over SIM for all vector-based baseline models. However, even the MAP method cannot match the performance of our two-step training regime: we report substantial gains with our FT-DIRECT and FT-MIXED fine-tuning on top of Reddit pretraining for all target domains but one (SEMEVAL15).

**Further Discussion.** The comparison of two fine-tuning strategies suggests that the simpler FT-DIRECT fine-tuning has an edge over FT-MIXED, and it seems that the gap between FT-DIRECT and FT-MIXED is larger on bigger datasets. However, as expected, FT-DIRECT adapts to the target task more aggressively: this leads to its degraded performance on the general-domain Reddit response selection task, see the scores in parentheses in Table 3. With more in-domain training data FT-DIRECT becomes worse on the REDDIT test set. On the other hand, FT-MIXED manages to maintain its high performance on REDDIT due to the in-batch mixing used in the fine-tuning process.[13]

**Qualitative Analysis.** The effect of fine-tuning is also exemplified by t-SNE plots for the BANK-

[13] Varying the parameter $M$ in FT-MIXED from the ratio 3:1 to 1:3 leads only to slight variations in the final results.

ING domain shown in Figure 4.[14] Recall that in our BANKING FAQ dataset several questions map to the same response, and ideally such questions should be clustered together in the semantic space. While we do not see such patterns at all with ELMO-encoded questions without mapping (ELMO-SIM, Figure 4a), such clusters can already be noticed with USE-MAP (Figure 4b) and with the model pretrained on Reddit without fine-tuning (Figure 4c). However, fine-tuning yields the most coherent clusters by far: it attracts encodings of all similar questions related to the same category closer to each other in the semantic space. This is in line with the results reported in Table 3.

## 5 Related Work

**Retrieval-Based Dialogue Systems.** Retrieval-based systems (Yan et al., 2016; Bartl and Spanakis, 2017; Wu et al., 2017; Song et al., 2018; Weston et al., 2018, *inter alia*) provide less variable output than generative dialogue systems (Wen et al., 2015, 2017a; Vinyals and Le, 2015), but they offer a crucial advantage of producing more informative, semantically relevant, controllable, and grammatically correct responses (Ji et al., 2014). Unlike modular and end-to-end task-oriented systems (Young, 2010; Wen et al., 2017b; Mrkšić and Vulić, 2018; Li et al., 2018), they do not require expensive curated domain ontologies, and bypass the modelling of complex domain-specific decision-making policy modules (Gašić et al., 2015; Chen et al., 2017). Despite these desirable properties, their potential has not been fully exploited in task-oriented dialogue.

Their fundamental building block is response selection (Banchs and Li, 2012; Wang et al., 2013; Al-Rfou et al., 2016; Baudis and Sedivý, 2016). We have witnessed a recent rise of interest in neural architectures for modelling response selection (Wu et al., 2017; Chaudhuri et al., 2018; Zhou et al., 2018; Tao et al., 2019), but the progress is still hindered by insufficient domain-specific training data (El Asri et al., 2017; Budzianowski et al., 2018). While previous work typically focused on a single domain (e.g., Ubuntu technical chats (Lowe et al., 2015, 2017)), in this work we show that much larger general-domain Reddit data can be leveraged to pretrain response selection models that support more specialised target dialogue domains.

[14]For clarity, we show the plots with 10 (out of 77) selected categories, while the full plots with all 77 categories are available in the supplemental material.

To the best of our knowledge, the work of Henderson et al. (2017) and Yang et al. (2018) is closest to our response selection pretraining introduced in §2.1. However, Henderson et al. (2017) optimise their model for one single task: replying to e-mails with short messages (Kannan et al., 2016). They use a simpler feed-forward encoder architecture and do not consider wide portability of a single general-domain response selection model to diverse target domains through fine-tuning. Yang et al. (2018) use Reddit conversational context to simply probe semantic similarity of sentences (Agirre et al., 2012, 2013; Nakov et al., 2016), but they also do not investigate response selection fine-tuning across diverse target domains.

**Pretraining and Fine-Tuning.** Task-specific fine-tuning of language models (LMs) pretrained on large unsupervised corpora (Peters et al., 2018; Devlin et al., 2019; Howard and Ruder, 2018; Radford et al., 2018; Lample and Conneau, 2019; Liu et al., 2019) has taken NLP by storm. Such LM-based pretrained models support a variety of NLP tasks, ranging from syntactic parsing to natural language inference (Peters et al., 2018; Devlin et al., 2019), as well as machine reading comprehension (Nishida et al., 2018; Xu et al., 2019) and information retrieval tasks (Nogueira and Cho, 2019; Yang et al., 2019). In this work, instead of the LM-based pretraining, we put focus on the response selection pretraining in particular, and show that such models coupled with target task fine-tuning (Howard and Ruder, 2018) lead to improved modelling of conversational data in various domains.

## 6 Conclusion and Future Work

We have presented a novel method for training neural response selection models for task-oriented dialogue systems. The proposed training procedure overcomes the low-data regime of task-oriented dialogue by pretraining the response selection model using general-domain conversational Reddit data and efficiently adapting this model to individual dialogue domains using in-domain data. Our evaluation demonstrates the compelling benefits of such pretraining, with the proposed training procedure achieving strong performance across each of the five different dialogue domains. In future work, we will port this approach to additional target domains, other languages, and investigate more sophisticated encoder architectures and fine-tuning strategies.

## References


Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 Task 6: A pilot on semantic textual similarity. In *Proceedings of *SEM*, pages 385–393.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *sem 2013 shared task: Semantic textual similarity. In *Proceedings of *SEM*, pages 32–43.

Rami Al-Rfou, Marc Pickett, Javier Snaider, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2016. Conversational contextual cues: The case of personalization and history for response ranking. *CoRR*, abs/1606.00372.

Rafael E. Banchs and Haizhou Li. 2012. IRIS: A chat-oriented dialogue system based on the vector space model. In *Proceedings of ACL System Demos*, pages 37–42.

Alexander Bartl and Gerasimos Spanakis. 2017. A retrieval-based dialogue system utilizing utterance and context embeddings. *CoRR*, abs/1710.05780.

Petr Baudis and Jan Sedivý. 2016. Sentence pair scoring: Towards unified framework for text comprehension. *CoRR*, abs/1603.06127.

Antoine Bordes, Y.-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *Proceedings of ICLR*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of EMNLP*, pages 5016–5026.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Debanjan Chaudhuri, Agustinus Kristiadi, Jens Lehmann, and Asja Fischer. 2018. Improving response selection in multi-turn dialogue systems by incorporating domain knowledge. In *Proceedings of CoNLL*, pages 497–507.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, and Phillipp Koehn. 2013. One billion word benchmark for measuring progress in statistical language modeling. In *Proceedings of INTERPSEECH*, pages 2635–2639.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *SIGKDD Explorations Newsletter*, 19(2):25–35.

Valeriu Codreanu, Damian Podareanu, and Vikram A. Saletore. 2017. Scale out for large minibatch SGD: Residual network training on ImageNet-1K with improved accuracy and reduced time to train. *CoRR*, abs/1711.04291.

Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Wenchao Du and Alan Black. 2018. Data augmentation for neural online chats response selection. In *Proceedings of the 2nd International Workshop on Search-Oriented Conversational AI*, pages 52–58.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: A corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of SIGDIAL*, pages 207–219.

Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Policy committee for adaptation in multi-domain spoken dialogue systems. In *Proceedings of ASRU*, pages 806–812.

Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *CoRR*, abs/1706.02677.

Mandy Guo, Qinlan Shen, Yinfei Yang, Heming Ge, Daniel Cer, Gustavo Hernandez Abrego, Keith Stevens, Noah Constant, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Effective parallel corpus mining using bilingual sentence embeddings. In *Proceedings of WMT*, pages 165–176.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of ICCV*, pages 1026–1034.

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

Matthew Henderson, Pawel Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019. A repository of conversational datasets. In *Proceedings of the 1st Workshop on Natural Language Processing for Conversational AI*.

Matthew Henderson, Blaise Thomson, and Jason D. Wiliams. 2014a. The Second Dialog State Tracking Challenge. In *Proceedings of SIGDIAL*, pages 263–272.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014b. Word-based dialog state tracking with recurrent neural networks. In *Proceedings of SIGDIAL*, pages 292–299.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of ACL*, pages 328–339.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of ACL*, pages 1681–1691.

Zongcheng Ji, Zhengdong Lu, and Hang Li. 2014. An information retrieval approach to short text conversation. *CoRR*, abs/1408.6988.

Anjuli Kannan, Karol Kurach, Sujith Ravi, Tobias Kaufmann, Andrew Tomkins, Balint Miklos, Greg Corrado, László Lukács, Marina Ganea, Peter Young, and Vivek Ramavajjala. 2016. Smart Reply: Automated response suggestion for email. In *Proceedings of KDD*, pages 955–964.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. *CoRR*, abs/1706.02515.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of NAACL-HLT*, pages 110–119.

Xiujun Li, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Celikyilmaz. 2017. End-to-end task-completion neural dialogue systems. In *Proceedings of IJCNLP*, pages 733–743.

Xiujun Li, Sarah Panda, Jingjing Liu, and Jianfeng Gao. 2018. Microsoft dialogue challenge: Building end-to-end task-completion dialogue systems. *CoRR*, abs/1807.11125.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of LREC*.

Fei Liu and Julien Perez. 2017. Gated end-to-end memory networks. In *Proceedings of EACL*, pages 1–10.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *CoRR*, abs/1901.11504.

Ryan Lowe, Nissan Pow, Iulian V. Serban, and Joelle Pineau. 2015. The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of SIGDIAL*, pages 285–294.

Ryan Thomas Lowe, Nissan Pow, Iulian Vlad Serban, Laurent Charlin, Chia-Wei Liu, and Joelle Pineau. 2017. Training end-to-end dialogue systems with the ubuntu dialogue corpus. *Dialogue & Discourse*, 8(1):31–65.

Laurens van der Maaten and Geoffrey E. Hinton. 2012. Visualizing non-metric similarities in multiple maps. *Machine Learning*, 87(1):33–55.

Yury A. Malkov and D. A. Yashunin. 2016. Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs. *CoRR*, abs/1603.09320.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Dominic Masters and Carlo Luschi. 2018. Revisiting small batch training for deep neural networks. *CoRR*, abs/1804.07612.

Nikola Mrkšić and Ivan Vulić. 2018. Fully statistical neural belief tracking. In *Proceedings of ACL*, pages 108–113.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. In *Proceedings of ACL*, pages 794–799.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Tsung-Hsien Wen, and Steve Young. 2017a. Neural Belief Tracker: Data-driven dialogue state tracking. In *Proceedings of ACL*, pages 1777–1788.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017b. Semantic specialisation of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the ACL*, pages 314–325.

Preslav Nakov, Lluís Màrquez, Walid Magdy, Alessandro Moschitti, Jim Glass, and Bilal Randeree. 2015. SemEval-2015 Task 3: Answer selection in community question answering. In *Proceedings of SEMEVAL*, pages 269–281.

Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, abed Alhakim Freihat, Jim Glass, and Bilal Randeree. 2016. SemEval-2016 Task 3: Community question answering. In *Proceedings of SEMEVAL*, pages 525–545.

Kyosuke Nishida, Itsumi Saito, Atsushi Otsuka, Hisako Asano, and Junji Tomita. 2018. Retrieve-and-read:

Multi-task learning of information retrieval and reading comprehension. In *Proceedings of CIKM*, pages 647–656.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with BERT. *CoRR*, abs/1901.04085.

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *CoRR*, abs/1701.06548.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. *Technical Report, OpenAI*.

Prajit Ramachandran, Barret Zoph, and Quoc V. Le. 2017. Searching for activation functions. *CoRR*, abs/1710.05941.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of Twitter conversations. In *Proceedings of NAACL-HLT*, pages 172–180.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Nicolas Schrading, Cecilia Ovesdotter Alm, Ray Ptucha, and Christopher Homan. 2015. An analysis of domestic abuse discourse on Reddit. In *Proceedings of EMNLP*, pages 2577–2583.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of AAAI*, pages 3776–3784.

Pararth Shah, Dilek Hakkani-Tür, Bing Liu, and Gokhan Tür. 2018. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of NAACL-HLT*, pages 41–51.

Samuel L. Smith, Pieter-Jan Kindermans, and Quoc V. Le. 2017. Don't decay the learning rate, increase the batch size. In *Proceedings of ICLR*.

Yiping Song, Cheng-Te Li, Jian-Yun Nie, Ming Zhang, Dongyan Zhao, and Rui Yan. 2018. An ensemble of retrieval-based and generation-based human-computer conversation systems. In *Proceedings of IJCAI*, pages 4382–4388.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of CVPR*, pages 2818–2826.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In *Proceedings of WSDM*, pages 267–275.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NIPS*, pages 6000–6010.

Oriol Vinyals and Quoc Le. 2015. A Neural Conversational Model. In *Proceedings of ICML Deep Learning Workshop*.

Mengting Wan and Julian McAuley. 2016. Modeling ambiguity, subjectivity, and diverging viewpoints in opinion question answering systems. In *Proceedings of ICDM*, pages 489–498.

Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. 2013. A dataset for research on short-text conversations. In *Proceedings of EMNLP*, pages 935–945.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of EMNLP*, pages 1711–1721.

Tsung-Hsien Wen, Yishu Miao, Phil Blunsom, and Steve J. Young. 2017a. Latent intention dialogue models. In *Proceedings of ICML*, pages 3732–3741.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017b. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of EACL*, pages 438–449.

Jason Weston, Emily Dinan, and Alexander Miller. 2018. Retrieve and refine: Improved sequence generation models for dialogue. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI*, pages 87–92.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of ACL*, pages 496–505.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. *CoRR*, abs/1904.02232.

Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In *Proceedings of SIGIR*, pages 55–64.

Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Simple applications of BERT for ad hoc document retrieval. *CoRR*, abs/1903.10972.

Yinfei Yang, Steve Yuan, Daniel Cer, Sheng-Yi Kong, Noah Constant, Petr Pilar, Heming Ge, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Learning semantic textual similarity from conversations. In *Proceedings of The 3rd Workshop on Representation Learning for NLP*, pages 164–174.

Steve Young. 2010. Still talking to machines (cognitively speaking). In *Proceedings of INTERSPEECH*, pages 1–10.

Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *Proceedings of ACL*, pages 1118–1127.