Domain Adaptation for Conversation Response Ranking

Gustavo Penha  
TU Delft  
g.penha-1@tudelft.nl

Claudia Hauff  
TU Delft  
c.hauflf@tudelft.nl

ABSTRACT
In this work we explore to what extent conversation response ranking models trained on some domains and applied to others (e.g., domains for which few or no training instances exist) can be improved when employing two—on first sight opposite—regularization approaches. We investigate the effectiveness of domain-aware and domain-agnostic representations obtained via regularization under two conditions: (i) domain granularity and (ii) regularization depth (i.e., at which network layer to incorporate the regularization loss). We conduct a comprehensive set of experiments on four dialogue datasets and focus on the out-of-domain effectiveness. We demonstrate that domain-aware representations consistently outperform domain-agnostic ones in a fine-grained domain setting (i.e., topics as domains), while domain-agnostic representations proves to be more effective in a coarse-grained domain setting (i.e., datasets as domains). With respect to the regularization depth, we find regularization in the initial layers to be more effective than regularization employed in later (i.e., deeper) layers.

KEYWORDS
multi-task learning, domain adversarial learning, domain adaptation, conversational response ranking, conversational search, information-seeking conversations

1 INTRODUCTION
In information-seeking dialogues, given a user utterance (and potentially additional context information such as the user’s profile and conversation history), the system response is either generated on the fly [25] or retrieved from a corpus of historical conversations [35]; hybrid approaches are beginning to appear too [22, 38]. While generated responses could, in principle, provide adequate replies to any utterance, in practice the models often generate uninformative responses [13] or responses that are incoherent given the conversation history [14]. In contrast, conversation response ranking relies on the existence of a large corpus of historical conversation data and adequate replies (that are coherent, well-formulated and informative) can be found in the historical data [28, 38].

In this paper, we focus on this very ranking problem. Neural approaches to conversation response ranking learn representations from a large number of training dialogues, in order to distinguish between relevant and irrelevant candidate responses to an utterance. Neural models often overfit to the domain (a data distribution) they were trained on, learning representations that are only useful on a specific domain while failing on instances from unseen domains [24, 31]. Additionally, there are domains where we have limited dialogue data to train on, and thus training effective neural models on specific domains may be infeasible. Our goal is to study regularization approaches to learn representations for conversation response ranking that generalize well to out-of-domain instances.

Recently, Cohen et al. [5] adapted domain adversarial learning (DAL) [8] to improve vanilla neural-net based passage-retrieval rankers, by learning representations that are agnostic to the domain of the search query. The proposed model outperformed the vanilla (non-regularized) neural ranking model on the passage retrieval task for out-of-domain queries. At the same time, Liu et al. [15] demonstrated significant improvements on the web search ranking task (compared to a model singularly trained for this task) when employing the multi-task learning (MTL) setup to achieve the opposite effect of domain-aware representations: apart from document ranking, the neural ranker was simultaneously trained for the task of query domain classification.

While both MTL and DAL have been shown to outperform a vanilla neural ranker, they have not yet been compared with each other in a unified ranking setup. It is still unclear why and when the domain is useful or not in neural ranking models’ representations. We empirically explore across four conversation datasets how MTL and DAL perform when employed on top of a common base model—Deep Matching Networks (DMN) [39]—that provides us with a strong baseline for conversation response ranking.

Our main findings are: (1) DMN+MTL consistently outperforms DMN and DMN+DAL in the fine-grained domain setting (topics such as travel and physics as domains), while DMN+DAL proves superior in the coarse-grained domain setting. We provide evidence for the hypothesis that this is caused by the strength of the domain shift [9, 30], i.e., the distribution differences between domains; and (2) regularization in the initial layers—that learn to represent the conversation history and the current utterance—is more effective than regularization employed in deeper layers that learns how to match the utterance with the conversation history.

2 BACKGROUND
Conversation Response Ranking. Early studies on retrieval-based dialogue systems focused on single-turn conversations [29, 37]. More recently, researchers have explored techniques for conversation response ranking [16, 35, 38, 39] which is the task of selecting a response from a set of response candidates using all previous turns in the conversation as input. This is significantly more complex than retrieval for single-turn interactions, as the ranking model has to determine where the important information is in the set of previous utterances. There are two main approaches in neural ranking models: representation-focused [27] and interaction-focused [10]. The former learns query (for our task this means all previous utterances in the conversation) and document (candidate response) representations separately and then computes the similarity between the representations. In the latter approach, first a query-document interaction matrix is built, which is then fed to neural net layers. Deep Matching Networks [35] (DMN for short) belong to the latter group,

The source code is available at https://github.com/Guzpenha/DomainRegularizedDeepMatchingNetworks
which has shown to outperform representation-focused approaches on several text matching tasks [19]. DMN was proposed to tackle the conversation response ranking task by matching the response candidates with all previous utterances in the dialogue and encoding the information in several interaction matrices, unlike previous models that either first represent all previous utterances as a vector and then match it to the response [16].

**Multi-Task Learning.** In MTL [6, 11, 17], we train a single model for different tasks at the same time in order to obtain effectiveness improvements, as compared to training models individually. Caruana [3] shows that semantically related tasks are required for MTL to work well. The sources of MTL’s improvements can be attributed to a range of factors [26] including choosing representations that other tasks might also prefer and acting as a regularizer. Liu et al. [15] demonstrated statistically significant improvements on the web search result ranking task when employing MTL in a two-task setup: result ranking and query domain classification. The resulting model has a representation that is able to distinguish between query domains (thus, is domain-aware) and outperforms a ranking model trained solely for ranking.

**Domain Adaptation.** The often observed discrepancy between training instances and test instances’ distributions in machine learning applications is known as dataset bias or domain shift [9, 26, 30] and has lead to research [7, 33, 34] on the problem of domain adaptation: adapting the training procedure so that the models generalize to instances from a different domain. Domain adaptation has only recently been studied in ranking tasks. Tran et al. [32] applied domain adaptation techniques, namely DAL [8] and Maximum Mean Discrepancy [9], in a learning to rank framework and showed effectiveness increases on the task of email search. The underlying strategy of DAL is to learn source and target representations that are as indistinguishable as possible through a domain classifier that works adversarially to the main objective by a gradient reversal layer. Cohen et al. [5] also applied DAL to neural ranking models and showed that it is effective for domain adaptation in the passage retrieval task compared to the non-adapted neural model.

**Study Motivation.** Our study is motivated by the following observations. We lack research on whether MTL or DAL representation regularizers are effective for conversation response ranking. Moreover, MTL and DAL are fundamentally opposing approaches (inducing domain-aware and domain agnostic representations) that can be compared within a single experimental framework as we will show in §4. This raises the question under which conditions and why DAL or MTL is more effective. Finally, previous work focused either on the domain adaptation problem [5, 32], measuring only out-of-domain effectiveness, or the in-domain effectiveness [15] whereas we provide a unified comparison of both. To our knowledge, only Adi et al. [1] compared MTL and DAL for the voice transcription task and found neither MTL nor DAL to show consistent improvements over standard neural models. It is still unclear the necessary circumstances for each technique to be effective and it remains an open question if they are effective for the conversation response ranking task.

### 3 CONVERSATION RESPONSE RANKING

Before providing details of our MTL and DAL setup (in Section 4), let us formally define the task of conversation response ranking. Let \( D = \{(U_1, r_1, y_1), \ldots, (U_N, r_N, y_N)\} \) be an information-seeking conversations data set consisting of \( N \) triplets: dialogue context, response candidate and response label. The dialogue context \( U_i \) is composed of the previous utterances \( \{u_1, u_2, \ldots, u_{\tau-1}\} \) at the turn \( \tau \) of the dialogue. The candidates \( r_i \) can be either the true response \( u_\tau \) (and thus \( y_i = 1 \)), or a negative sampled candidate \( y_i = 0 \). The task is then to learn a function that is able to generate a ranked list for a given set of candidate responses based on their retrieval scores \( f(U_i, r_i) \).

When discussing the domain adaptation problem, we refer to \( X \) as the input space (context and candidate response) and to \( Y \) as the output space (relevance score). We assume that we have \( s + t \) different distributions over \( X \times Y \), called source domains \( D_S = \{D_1, \ldots, D_s\} \) and target domains \( D_T = \{D_1, \ldots, D_t\} \). The domain adaptation task is to build a model \( f : X \rightarrow Y \) with high out-of-domain effectiveness (test instances from \( D_T \)), based on labeled source samples \( (x_i, y_i) \) drawn from source domains \( D_S \) and unlabeled target samples \( (x_i) \) from target domains \( D_T \).

### 4 DOMAIN-REGULARIZED DMN

We now introduce the components of our method, first DMN [39], followed by the DAL and MTL regularizers. As seen in Figure 1, there are two modules: (i) the DMN module, and, (ii) the domain classifier module. DMN accumulates the matching scores \( f \) between \( U_i, r_i \). The domain classifier module acts as domain regularizer and comes in two variants: inducing either domain-agnostic representations via DAL or domain-aware representations via MTL.

#### 4.1 DMN

**Context and Response Representations.** First, each utterance in the context and the candidate response are represented at the word and sentence level. The utterance word level representation \( E(u) \) is the concatenation of embedding vectors obtained from a global look-up embedding matrix, such as word embeddings [18], for every word in the utterance. The sentence level representation of each word is the concatenation of the forward and backward BiGRU [4] recurrent units which processes \( x_i \) in opposite directions. The BiGRU hidden states for each word are then concatenated \( h_{u\tau} = [\widehat{h}_{u\tau}, \overwidehat{h}_{u\tau}] \). We refer to the concatenation of all the utterance’s word hidden states \( h_{u\tau} \) as the sentence level representation \( H(u) \). The same process is applied to obtain response word level representation \( E(r) \) and sentence level representation \( H(r) \).

**Interaction Matching Matrices.** Two interaction matrices are then created: a word interaction matrix and a sentence interaction matrix. Matrix \( M_1 \) is defined as the word dot product similarity between the utterance word embedding representation \( E(u) \) and the candidate response word embedding representation \( E(r) \). It captures how similar the utterance and response are in terms of their word embeddings. Matrix \( M_2 \) is defined as the dot product similarity between the sentence level representation of the utterance \( H(u) \) and the sentence level representation of the response \( H(r) \).

**Matching Accumulation, Prediction and Training.** For each turn in the conversation (up to a certain window size \( c \), a hyperparameter
of the model), M1 and M2 are fed into a convolution layer, followed by a max pooling layer, in order to learn higher level matching patterns. This generates a matrix for each utterance/candidate-response pair in the conversation so far; those outputs are feed into a BiGRU layer that accumulates the results of the max pooling layer into hidden states $H_{acc} = [h_1, ..., h_L]$, which are concatenated and fed into a fully connected layer (FCL) that determines the final matching score: $f(U_i, r_j) = \sigma(w_{f}^T \cdot \text{tanh}(w_{d}^T \cdot H_{acc} + b_{f}) + b_{d})$. Here, $\sigma$ is the softmax function and $w_f, w_d, b_f, b_d$ are model parameters.

To train DMN we use the pairwise hinge loss function. Each training instance is a triplet $(U_i, r_j, r_N)$, where $r_N$ is the true response and $r_j$ a negative sampled response. Formally, the relevance loss function is: $L_{\text{relevance}}(D, \theta_f, \theta_g) = \sum_{i=1}^{N} \max(0, \epsilon - f(U_i, r_j) + f(U_i, r_N))$, where $\theta_f$ and $\theta_g$ denote all the parameters of the model, $\epsilon$ is the margin for the hinge loss and $N$ is the number of triplets in the training data $D$. Having formally defined the base model, we now introduce the two regularizers: DMN+DAL and DMN+MTL.

### 4.2 DMN+DAL: Domain Adversarial Learning

To tackle domain adaptation, Ganin et al. [8] proposed to control the $\mathcal{H}$-divergence—a notion of distance between source and target domains proposed by Ben-David et al. [2]—by learning source and target representations that are as indistinguishable as possible through a domain classifier resulting in domain-agnostic representations. This classifier works adversarially to the main objective by adding a gradient reversal layer between the domain classifier and the model representation layers. Inspired by [5], we implement this approach for DMN. As seen in Figure 1, there are a set of shared weights $\theta_f$ regarding the textual representation and matching layers of the network, that are used by both the domain classifier (a fully connected layer with its own set of weights $\theta_d$), and the final matching scorer (a fully connected layer with its own set of weights $\theta_g$). During network training, the domain loss backpropagates (red arrow) through the domain classifier weights ($\theta_f$) and after that a gradient reversal layer (GRL) flips the sign of the gradient. This procedure is known as domain adversarial learning and can be represented by another term in the loss function (the domain loss) that is subtracted from the relevance loss:

$$L_{\text{DAL}} = L_{\text{relevance}}(D, \theta_f, \theta_g) + \lambda \cdot L_{\text{domain}}(D, \theta_f, \theta_d)$$

Note that Equation 1 showcases both DAL and MTL, the latter is explained in the next section. For both approaches, the domain classifier is a fully connected layer that can take its input from various points of the network layers—we here explore two points at different network depths. The input for the domain classifier is thus either the concatenation of the textual representations of both the context and response candidate $[E(u), H(u), E(r), H(r)]$ (Depth I in Figure 1) or the accumulated matching score $H_{acc}$ (Depth II in Figure 1). The activation function of the domain classifier is a softmax, and the loss function used for training is the categorical cross entropy.

### 4.3 DMN+MTL: Multi-Task Learning

According to the MTL line of reasoning, the tasks of conversation domain classification and conversation response ranking can benefit from a unified representation, as they are intrinsically related—and this relatedness of tasks is important for MTL to work well [3]. Identifying the domain of an information need (expressed here through utterances) is likely important for ranking the correct response highly [15]. In our MTL setup, we keep almost our entire
We consider four information-seeking conversation datasets\footnote{We use their default train, development and test splits.} that have been used in prior works\cite{16,21,35,38}:

- **MSDialog**\footnote{MSDialog is available at https://citr.cs.umass.edu/downloads/msdialog/}: a total of 246K context-response pairs, built from 35.5K information seeking conversations from the Microsoft Answer community, a QA forum for several Microsoft products.
- **UDC**\footnote{UDC is available at https://www.dropbox.com/s/2fdn26rj6h9bpvl/ubuntu%20data.zip}: the Ubuntu Dialog Corpus contains 2 million context-response pairs collected from the chat logs of the IRC network concerning technical support (Ubuntu) conversations.
- **MANtIS**\footnote{MANtIS is available at https://ciir.cs.umass.edu/downloads/msdialog/}: 1.3 million context-response pairs built from conversations of 14 diverse sites of Stack Exchange.
- **SEApplex\footnote{SEApps is available at https://drive.google.com/open?id=1gPUmAqv7_zjIlJwP07h1ad8RoUL7IeB}: 258k context-response pairs collected from Stack Exchange\footnote{https://apple.stackexchange.com/, dump of 2019-03-04}, containing conversations about Apple products. This is a subset of MANtIS we created to provide a third technical only domain dataset (in addition to the popular MSDialog and UDC datasets).

### 5.1 Datasets

We differ by setting $c = 2$ previous utterances for accumulating matching scores and set the maximum possible utterance length to 30 words, achieving close DMN results while increasing efficiency. Given the large number of models we train—26 (combinations of source and target domains) $\times$ 5 (DMN variations) $\times$ 5 (different random seeds) = 650—we reduced the number of iterations to 20% of the ones reported by\cite{39}, which reduced the MAP in $\sim$ 20% (cf. Table 3 train on MS and test on MS). We employed word2vec\cite{18} with 200 dimensions and pretrain them on both the source and target datasets in all our experiments; the embeddings are then fine-tuned during the training of the models. We implemented all models with TensorFlow on top of the publicly available DMN implementation\footnote{https://github.com/yangliuy/NeuralResponseRanking}. The models were trained using Adam optimizer\cite{12} with an initial learning rate of 0.001.

### 6 EXPERIMENTAL RESULTS

The central results of our experiments are shown in Tables 2 and 3. Let us briefly describe how to read the tables’ result rows, using the first row of Table 2 as a concrete example. Here, we train our five model variants on the training splits of all domains within MANtIS apart from the domain apple (and thus $D_S = all \setminus apple$). We then report the effectiveness (in MAP) on the test splits of the domain $D_T = apple$ (that is the out-of-domain effectiveness) and the test splits of $D_T = all \setminus apple$ (the in-domain effectiveness).

#### 6.1 Domain Granularity

**Fine-grained: Topics Level.** The MSDialog (75 topics, such as Outlook, Bing Search and MSN) and MANtIS (14 topics such as electronics, travel and physics) datasets contain one topic label for each conversation and are thus suitable for our topic-level experiments. We present the test set results for the ten most frequently discussed topics per dataset in Table 2. We report the effectiveness (in MAP, averaged over five runs) of our DMN baseline, and the four regularization variants DMN+DAL$^I$ (DAL with input depth I), DMN+DAL$^{II}$ (DAL with input depth II), DMN+MTL$^I$ and DMN+MTL$^{II}$. Our train/test dataset combinations are either in-domain (i.e., we train and test on the same topics) or out-of-domain (i.e., we train on all but topic T and test on conversations with topic T).
Table 2: MAP results considering topics inside each dataset as domains, average of 5 runs for each model with different random initialization weights. Bold values indicate the highest values for each line. DMN is the baseline model and all subsequent columns with header +X^T should be read as DMN+X^T, where X is the regularization type and Y is the regularization depth.

<table>
<thead>
<tr>
<th>train on D_S</th>
<th>test on D_T (out-of domain MAP)</th>
<th>test on D_S (in-domain MAP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_T</td>
<td>DMN + DAL^I + DAL^II + MTL^I + MTL^II</td>
<td>D_S</td>
</tr>
<tr>
<td>all \ apple</td>
<td>apple</td>
<td>all \ apple</td>
</tr>
<tr>
<td>all \ dba</td>
<td>dba</td>
<td>all \ dba</td>
</tr>
<tr>
<td>all \ physics</td>
<td>physics</td>
<td>all \ physics</td>
</tr>
<tr>
<td>all \ english</td>
<td>English</td>
<td>all \ english</td>
</tr>
<tr>
<td>all \ security</td>
<td>Security</td>
<td>all \ security</td>
</tr>
<tr>
<td>all \ gaming</td>
<td>Gaming</td>
<td>all \ gaming</td>
</tr>
<tr>
<td>all \ gis</td>
<td>Gis</td>
<td>all \ gis</td>
</tr>
<tr>
<td>all \ askubuntu</td>
<td>Askubuntu</td>
<td>all \ askubuntu</td>
</tr>
<tr>
<td>all \ stats</td>
<td>Stats</td>
<td>all \ stats</td>
</tr>
</tbody>
</table>

In most cases (34 out of 40 across both datasets and in-domain and out-of-domain instances), we find DMN+MTL^I to perform best, with an average improvement over DMN of 2.8%. The DAL variants consistently perform worse than DMN. This finding is in contrast to [5] who report across their topic-level experiments DAL to outperform their vanilla neural net approach. We note though that their results are based on a different ranking task (passage retrieval) using different neural ranking models. At the same time, our cross-topics results are in agreement with prior evidence [15] that learning domain-aware query representations (i.e., MTL-based) are effective for ranking, considering in domain instances, and, surprisingly for out of domain instances as well.

Thus, in the granularity of topics as domain, we observe that inducing domain-aware representations are the most effective regularization technique. We argue that DMN+DAL is not effective in this scenario due to the domain shift, i.e. the difference between the source and target distributions, not being overly strong. Consider the left plot of Figure 2a. It shows a t-SNE visualization of the utterances sentence representation using a trained DMN. Target domain instances are displayed with ●, while source instances are colored ○. We note that the target and source instances are similarly spread throughout this two dimensional space. This suggests that the source and target distributions are quite similar. Since the high-level goal of domain adaptation is to tackle the domain shift problem by pushing the two distributions closer, the already existent overlap between source and target distributions in DMN representations could be resulting in the DMN+DAL ineffectiveness. Moreover, one may expect that if the learnt source and target representations are similar, a model trained on the source domain to generalize to the target domain. Comparing the in-domain

Table 3: MAP considering datasets as domains, average of 5 runs for each model with different random initialization weights. Bold values indicate the highest values for each line. DMN is the baseline model and all subsequent columns with header +X^T should be read as DMN+X^T, where X is the regularization type and Y is the regularization depth.

<table>
<thead>
<tr>
<th>train on D_S</th>
<th>test on D_T (out-of domain MAP)</th>
<th>test on D_S (in-domain MAP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_T</td>
<td>DMN + DAL^I + DAL^II + MTL^I + MTL^II</td>
<td>D_S</td>
</tr>
<tr>
<td>MS</td>
<td>UDC</td>
<td>MS</td>
</tr>
<tr>
<td>MS</td>
<td>Apple</td>
<td>MS</td>
</tr>
<tr>
<td>Apple</td>
<td>MS</td>
<td>Apple</td>
</tr>
<tr>
<td>Apple</td>
<td>UDC</td>
<td>Apple</td>
</tr>
<tr>
<td>UDC</td>
<td>Apple</td>
<td>UDC</td>
</tr>
</tbody>
</table>

In most cases (34 out of 40 across both datasets and in-domain and out-of-domain instances), we find DMN+MTL^I to perform best, with an average improvement over DMN of 2.8%. The DAL variants consistently perform worse than DMN. This finding is in contrast to [5] who report across their topic-level experiments DAL to outperform their vanilla neural net approach. We note though that their results are based on a different ranking task (passage retrieval) using different neural ranking models. At the same time, our cross-topics results are in agreement with prior evidence [15] that learning domain-aware query representations (i.e., MTL-based) are effective for ranking, considering in domain instances, and, surprisingly for out of domain instances as well.

Thus, in the granularity of topics as domain, we observe that inducing domain-aware representations are the most effective regularization technique. We argue that DMN+DAL is not effective in this scenario due to the domain shift, i.e. the difference between the source and target distributions, not being overly strong. Consider the left plot of Figure 2a. It shows a t-SNE visualization of the utterances sentence representation using a trained DMN. Target domain instances are displayed with ●, while source instances are colored ○. We note that the target and source instances are similarly spread throughout this two dimensional space. This suggests that the source and target distributions are quite similar. Since the high-level goal of domain adaptation is to tackle the domain shift problem by pushing the two distributions closer, the already existent overlap between source and target distributions in DMN representations could be resulting in the DMN+DAL ineffectiveness. Moreover, one may expect that if the learnt source and target representations are similar, a model trained on the source domain to generalize to the target domain. Comparing the in-domain
Coarse-grained: Datasets Level. Table 3 contains the results of our dataset-level experiments for which we employed the three technical support datasets MSDialog, UDC, and SEApple. The out-of-domain experiments, DMN+DAL fails for four out of six dataset combinations (average improvement of 6.7% over DMN), whereas in the in-domain setting the baseline is most effective.

In the two dataset combinations that DMN+DAL fails (first and fourth rows), we note that all other variants and the baseline also fail to generalize from a source domain (MSDialog in first row and SEApple in the fourth row) to the UDC domain: if we do not train DMN (i.e., randomly initialize its weights without further training) our untrained DMN also achieves a MAP of 0.3. This difficulty to generalize to UDC could mean that its data distribution is too different to begin with: while MSDialog and SEApple have similar average words per utterance scores (55.8 and 69.3 respectively) and average number of turns (3.5 and 5.0 respectively), UDC has shorter utterances (12.1 average words) and longer dialogues (10.1 turns).

Figure 2b (left) shows the representation of utterances in SEApple and MSDialog for DMN: the embeddings are not identical, but have some overlap. Those seem to be necessary conditions for domain adaptation techniques to work [32]. To conclude, in the granularity of datasets as domains, we find domain-agnostic representations to be the most effective regularization technique.

6.2 Regularization Depth

Across all experiments we reported so far, we can observe that inducing domain variance (MTL) and invariance (DAL) respectively, is more successful when employing representations of input depth I (i.e., word and sentence embeddings of the utterances) instead of input depth II (i.e., representations based on matching matrices). This result is somewhat expected since the matching representations capture the similarity scores between the utterances in the context and the response, i.e. how semantically similar the response is to each of the previous utterances, and the convolution and pooling layers act as local filters, learning where is important to match: word/sentence wise and turn-wise. Thus, the word information is present in the representation of earlier layers, while deeper layers capture how such representations from utterances and response candidate match.

7 CONCLUSIONS

In this work we have compared two (seemingly opposing) regularization approaches for the task of conversational response ranking: multi-task learning for domain classification and domain adversarial learning. Our results show that different domain regularization techniques for deep matching networks work under different set of conditions: (i) DMN+MTL is most effective when the domain granularity is high or fine-grained (topic-level experiments), whereas DMN+DAL is more effective at coarser domain granularities (dataset-level experiments) likely due to the existing differences in the strength of the domain shift and (ii) applying regularization at deeper layers of matching scores (input depth II) in the network is less effective than regularizing based on the early layers (input depth I) of textual representation.

Based on our findings, in future work we will explore the following three avenues: (i) Although we have found domain regularization to lead to effectiveness improvements, overall, the models’ effectiveness is not yet high enough to be suitable for actual retrieval-based open-domain conversational systems. We rely on a number of good response candidates (in line with all prior works [16, 35, 36]) and it is still an open question how to create a good response pool in the first place, (ii) we generated MANtIS from Stack Exchange, considering the sub-portals that are neither too popular nor too small. Overall, Stack Exchange hosts more than 170 sub-portals, and thus large-scale experiments across all domains will allow us to consider additional factors such as user types, domain clusters and so on.

9Due to computational constraints, we opted for a smaller set of initial domains in this work.
ACKNOWLEDGMENTS

This research has been supported by NWO projects SearchX (639.022.722) and NWO Aspasia (015.013.027).

REFERENCES


