Domain Adaptation for Conversation Response Ranking

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

Claudia Hauff
TU Delft
c.hauff@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 [1] 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,

[1] 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 $\mathcal{D} = \{(\mathcal{U}_i, r_i, y_i)\}_{i=1}^{N}$ be an information-seeking conversations data set consisting of N triplets: dialogue context, response candidate and response label. The dialogue context $\mathcal{U}_i$ is composed of the previous utterances $\{u^1, u^2, ..., u^{\tau-1}\}$ at the turn τ of the dialogue. The candidates r_i can be either the true response u_i^τ (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 $f(\mathcal{U}_i, r_i)$.

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 $\mathcal{D}_S = \{\mathcal{D}_1, ..., \mathcal{D}_s\}$ and target domains $\mathcal{D}_T = \{\mathcal{D}_1, ..., \mathcal{D}_t\}$. The domain adaptation task is to build a model $f : X \rightarrow Y$ with high out-of-domain effectiveness (test instances from $\mathcal{D}_T$), based on labeled source samples (x_i, y_i) drawn from source domains $\mathcal{D}_S$ and unlabeled target samples (x_i) from target domains $\mathcal{D}_T$.

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 $\mathcal{U}_i, r_i$. 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 $E(u)$ in opposite directions. The BiGRU hidden states for each word are then concatenated $h_{u_{wi}} = [\overrightarrow{h}_{u_{wi}}, \overleftarrow{h}_{u_{wi}}]$. We refer to the concatenation of all the utterance's word hidden states $h_{u_{wi}}$ 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 $M1$ 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 $M2$ 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

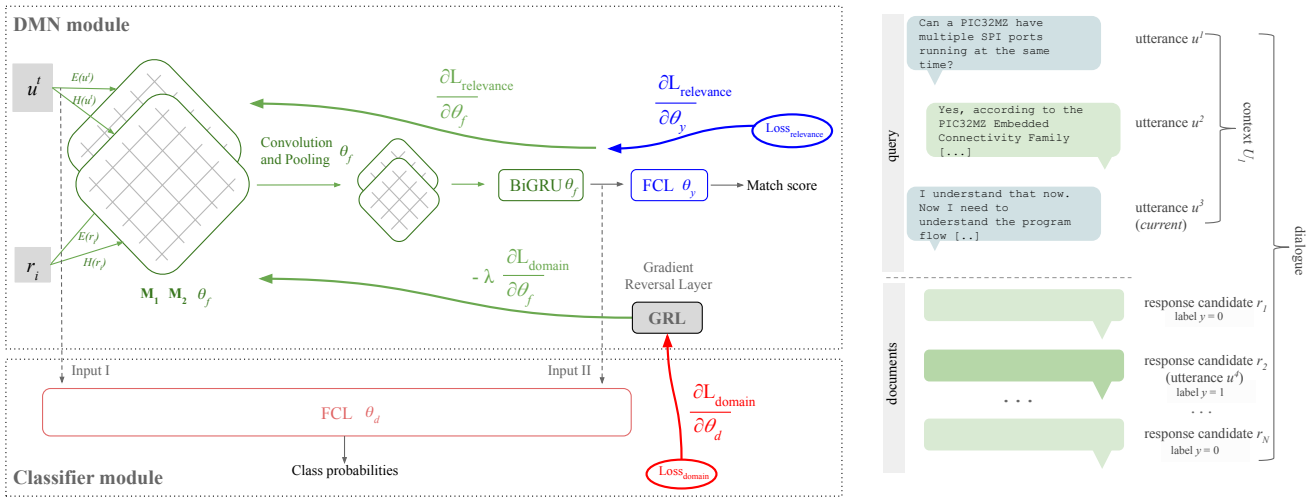


Figure 1: An overview of DMN and the domain regularizer. The green layers learn inputs representations, which is used by the blue layers to output a matching score and rank the responses (DMN). The regularization techniques (denoted by DMN+DAL or DMN+MTL) share the green layers of DMN to classify the dialogue into domains (red). The only difference between DMN+DAL and DMN+MTL is the inclusion of the gradient reversal layer (GRL). On the right we have a conversation example with the notation used throughout the paper.

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 $H_{acc} = [h_1, ..., h_c]$, which are concatenated and fed into a fully connected layer (FCL) that determines the final matching score: $f(\mathcal{U}_i, r_i) = \sigma(w_2^T \cdot tanh(w_1^T \cdot H_{acc} + b_1) + b_2)$. Here, σ is the softmax function and w_1, w_2, b_1, b_2 are model parameters.

To train DMN we use the pairwise hinge loss function. Each training instance is a triplet $(\mathcal{U}_i, r_i^+, r_i^-)$, where r_i^+ is the true response and r_i^- a negative sampled response. Formally, the relevance loss function is: $\mathcal{L}_{\text{relevance}}(\mathcal{D}, \theta_f, \theta_y) = \sum_{i=1}^{N} \max(0, \epsilon - f(\mathcal{U}_i, r_i^+) + f(\mathcal{U}_i, r_i^-))$, where θ_f and θ_f denote all the parameters of the model, ϵ is the margin for the hinge loss and N is the number of triplets in the training data $\mathcal{D}$. 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 θ_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 θ_d), and the final matching scorer (a fully connected layer with its own set of weights θ_y). During network training, the domain loss backpropagates (red arrow) through the domain classifier weights (θ_d) 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:

$$\mathcal{L}_{\text{DAL}\ \text{MTL}} = \mathcal{L}_{\text{relevance}}(\mathcal{D}, \theta_f, \theta_y) \ \text{DAL: } - \ \text{MTL: } + \ \lambda \cdot \mathcal{L}_{\text{domain}}(\mathcal{D}, \theta_f, \theta_d) \tag{1}$$

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

DAL setup intact, with the exception of the loss function. In order to achieve representations that are able to distinguish between domains, we flip the sign of the DAL loss function (cf. Equation 1).

5 EXPERIMENTAL SETUP

5.1 Datasets

We consider four information-seeking conversation datasets[2] that have been used in prior works [16, 21, 35, 38]:

`MSDialog`[3] [23]: 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`[4] [16]: 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`[5] [20]: 1.3 million context-response pairs built from conversations of 14 diverse sites of Stack Exchange.

`SEApple`[6]: 258k context-response pairs collected from Stack Exchange[7], 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)

Table 1: Statistics of the datasets used. $\mathcal{U}$ indicates the context, r response and u utterances.

	MSDialog			**UDC**		
has topics	yes			no		
# topics	75			-		
set	Train	Valid	Test	Train	Valid	Test
# ($\mathcal{U}$, r) pairs	173k	37k	35k	1000k	500k	500k
# cand. per $\mathcal{U}$	10	10	10	2	10	10
Avg # turns	5.0	4.8	4.4	10.1	10.1	10.1
Avg # words per u	55.8	55.8	52.7	7.3	7.3	7.3
Avg # words per r	67.3	68.8	67.7	12.1	12.1	12.1

	SEApple			**MANtIS**		
has topics	no			yes		
# topic	-			14		
set	Train	Valid	Test	Train	Valid	Test
# ($\mathcal{U}$, r) pairs	188k	33k	37k	904k	199k	197k
# cand. per $\mathcal{U}$	10	10	10	11	11	11
Avg # turns	3.5	3.7	3.7	4.0	4.1	4.1
Avg # words per u	69.3	84.3	84.3	98.2	107.2	110.4
Avg # words per r	63.3	79.9	83.0	91.0	100.1	94.6

5.2 Evaluation

We consider two evaluation schemes in our experiments: *in-domain* effectiveness and *out-of-domain* effectiveness. The out-of-domain evaluation measures how well the model performs for the domain adaptation task, i.e. test sets from the target domains $\mathcal{D}_T$. The in-domain effectiveness demonstrates how well the adapted models perform on the instances from the domain they were trained on, i.e. test sets from source domains $\mathcal{D}_S$.

[2]We use their default train, development and test splits.
[3]`MSDialog` is available at https://ciir.cs.umass.edu/downloads/msdialog/
[4]`UDC` is available at: https://www.dropbox.com/s/2fdn26rj6h9bpvl/ubuntu%20data.zip.
[5]`MANtIS` is available at https://guzpenha.github.io/MANtIS/.
[6]`SEApple` is available at https://drive.google.com/open?id=1gPUmAqv7_l2J2wF07h71adRoRaLITseB
[7]https://apple.stackexchange.com/, dump of 2019-03-04

We report the effectiveness on the test sets with respect to Mean Average Precision (MAP) similar to prior works [35, 39]. We train every model five times and report the average effectiveness over those five models in order to obtain more reliable evaluation values. We observe a low standard deviation (maximum of 0.004 MAP) and consistent results among different runs.

5.3 Implementation Details

Both DAL and MTL have only one hyperparameter, λ, that controls how strong the effect of the regularizer is when training the network. We gradually increase λ from 0 to 1 during the training process, using the following formula in all of our experiments: $\lambda_p = \frac{2}{1+\exp(-\gamma \cdot p)} - 1$, where p is the percentage of total iterations so far (it reaches 100% at the end of the training procedure) and γ was set to 10, the same strategy and γ employed by [8].

The DMN hyperparameters are the same as Yang et al. [39]. 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) * 5 (DMN variations) * 5 (different random seeds) = 650—we reduced the number of iterations to 20% of the ones reported by [39], which reduced the MAP in ~ 20% (cf. Table 3 train on MS and test on MS). We employed word2vec [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 [8]. The models were trained using Adam optimizer [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 $\mathcal{D}_S = all \setminus apple$). We then report the effectiveness (in MAP) on the *test splits* of the domain $\mathcal{D}_T = apple$ (that is the out-of-domain effectiveness) and the test splits of $\mathcal{D}_T = all \setminus apple$ (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).

[8]https://github.com/yangliuy/NeuralResponseRanking

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 $+X^Y$ should be read as DMN$+X^Y$, where X is the regularization type and Y is the regularization depth.

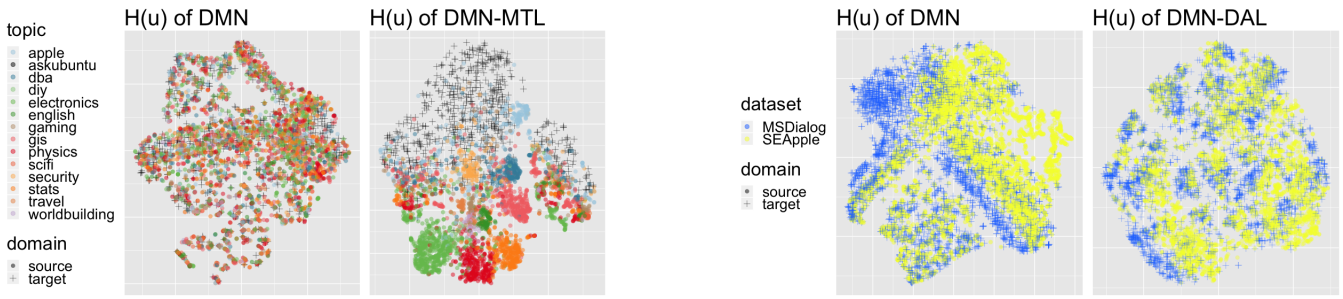
	MANtIS											
train on $\mathcal{D}_S$	test on $\mathcal{D}_T$ (out-of domain MAP)						test on $\mathcal{D}_S$ (in-domain MAP)					
	$\mathcal{D}_T$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}	$\mathcal{D}_S$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}
all \ apple	apple	0.512	0.494	0.270	**0.539**	0.491	all \ apple	0.514	0.503	0.276	**0.543**	0.488
all \ electronics	electronics	0.527	0.525	0.274	**0.528**	0.464	all \ electronics	**0.502**	**0.502**	0.271	0.499	0.493
all \ dba	dba	0.541	0.532	0.318	**0.564**	0.517	all \ dba	0.516	0.511	0.308	**0.534**	0.490
all \ physics	physics	0.544	0.528	0.436	**0.552**	0.522	all \ physics	0.510	0.500	0.447	**0.527**	0.501
all \ english	english	0.492	0.511	0.496	**0.514**	0.508	all \ english	0.504	0.501	0.493	**0.542**	0.504
all \ security	security	**0.564**	0.533	0.485	**0.564**	0.493	all \ security	0.526	0.500	0.471	**0.533**	0.474
all \ gaming	gaming	0.523	0.514	0.276	**0.545**	0.488	all \ gaming	0.512	0.505	0.275	**0.536**	0.489
all \ gis	gis	0.470	0.465	0.440	**0.485**	0.447	all \ gis	0.513	0.508	0.459	**0.534**	0.504
all \ askubuntu	askubuntu	0.474	0.464	0.368	**0.486**	0.451	all \ askubuntu	0.535	0.528	0.427	**0.548**	0.518
all \ stats	stats	0.533	0.527	0.279	**0.546**	0.455	all \ stats	0.504	0.499	0.272	**0.533**	0.479
	MSDialog											
train on $\mathcal{D}_S$	test on $\mathcal{D}_T$ (out-of domain MAP)						test on $\mathcal{D}_S$ (in-domain MAP)					
	$\mathcal{D}_T$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}	$\mathcal{D}_S$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}
all \ MSN	MSN	0.519	0.470	0.324	**0.530**	0.461	all \ MSN	0.541	0.482	0.322	**0.549**	0.484
all \ Onedrive	Onedrive	0.527	0.475	0.340	**0.539**	0.423	all \ Onedrive	0.547	0.490	0.344	**0.551**	0.485
all \ IE7	IE7	**0.533**	0.411	0.304	0.527	0.463	all \ IE7	0.546	0.423	0.303	**0.554**	0.481
all \ Windows7	Windows7	0.507	0.409	0.325	**0.520**	0.464	all \ Windows7	**0.564**	0.420	0.328	0.547	0.499
all \ Outlook	Outlook	**0.535**	0.461	0.316	0.529	0.421	all \ Outlook	**0.556**	0.458	0.314	0.551	0.500
all \ Band	Band	0.520	0.467	0.371	**0.535**	0.480	all \ Band	0.551	0.483	0.356	**0.558**	0.499
all \ Defender	Defender	0.532	0.406	0.361	**0.540**	0.490	all \ Defender	0.560	0.395	0.341	**0.564**	0.482
all \ Office	Office	0.512	0.479	0.344	**0.529**	0.455	all \ Office	0.547	0.498	0.343	**0.556**	0.491
all \ Lumia	Lumia	0.537	0.427	0.341	**0.542**	0.484	all \ Lumia	0.559	0.415	0.332	**0.562**	0.494
all \ OutlookIns	OutlookIns	0.509	0.431	0.340	**0.541**	0.448	all \ OutlookIns	0.530	0.445	0.338	**0.550**	0.496

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 $+X^Y$ should be read as DMN$+X^Y$, where X is the regularization type and Y is the regularization depth.

train on $\mathcal{D}_S$	test on $\mathcal{D}_T$ (out-of domain)						test on $\mathcal{D}_S$ (in-domain)					
	$\mathcal{D}_T$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}	$\mathcal{D}_S$	DMN	+ DAL^I	+ DAL^{II}	+ MTL^I	+ MTL^{II}
MS	UDC	0.289	0.292	0.293	0.284	**0.295**	MS	**0.534**	0.436	0.310	0.524	0.469
MS	Apple	0.455	**0.484**	0.283	0.418	0.397	MS	**0.529**	0.394	0.292	0.517	0.460
Apple	MS	0.325	**0.371**	0.296	0.325	0.338	Apple	0.657	0.600	0.290	**0.662**	0.649
Apple	UDC	0.298	0.297	0.303	0.309	**0.311**	Apple	**0.659**	0.609	0.381	0.647	0.644
UDC	Apple	0.356	**0.363**	0.291	0.314	0.358	UDC	**0.657**	0.629	0.295	0.645	0.646
UDC	MS	0.365	**0.382**	0.290	0.325	0.343	UDC	**0.658**	0.613	0.292	0.655	0.650

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



(a) Topic level **(b) Dataset level**

Figure 2: t-SNE visualization of the sentence representations $H(u)$ of the test instances. Shown are both the source domain (•) and the target domain (+). (a): topic-level setup of DMN and DMN+MTL^I (MANtIS topics); (b): dataset-level setup of DMN and DMN+DAL^I.

and out-of-domain MAP of DMN on Tab. 2 we see that the effectiveness drops only by an average of 4.8% MAP on MSDialog, whereas for MANtIS it actually increases by an average of 0.9 % MAP, which showcases that the domain shift is not strong.

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^I performs best 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^I 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[9].

[9]Due 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

[1] Yossi Adi, Neil Zeghidour, Ronan Collobert, Nicolas Usunier, Vitaliy Liptchinsky, and Gabriel Synnaeve. 2019. To Reverse the Gradient or Not: an Empirical Comparison of Adversarial and Multi-task Learning in Speech Recognition. In *ICASSP*. 3742–3746.
[2] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2007. Analysis of representations for domain adaptation. In *NeurIPS*. 137–144.
[3] Rich Caruana. 1997. Multitask learning. *Machine learning* 28, 1 (1997), 41–75.
[4] Kyunghyun Cho, Bart van Merrienboer Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *EMNLP*. 1724–1734.
[5] Daniel Cohen, Bhaskar Mitra, Katja Hofmann, and W Bruce Croft. 2018. Cross domain regularization for neural ranking models using adversarial learning. In *SIGIR*. 1025–1028.
[6] Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*. 160–167.
[7] Hal Daume III. 2007. Frustratingly Easy Domain Adaptation. In *ACL*. 256–263.
[8] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *JMLR* 17, 1 (2016), 2096–2030.
[9] A. Gretton, AJ. Smola, J. Huang, M. Schmittfull, KM. Borgwardt, and B. Schölkopf. 2009. *Covariate shift and local learning by distribution matching*. MIT Press, 131–160.
[10] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *CIKM*. 55–64.
[11] Lukasz Kaiser, Aidan N Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. 2017. One model to learn them all. *arXiv preprint arXiv:1706.05137* (2017).
[12] Diederik Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *ICLR* (12 2014).
[13] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A Diversity-Promoting Objective Function for Neural Conversation Models. In *NAACL*. 110–119.
[14] Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A Persona-Based Neural Conversation Model. In *ACL*. 994–1003.
[15] Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. 2015. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. In *NAACL*. 912–921.
[16] 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 *SIGDIAL*. 285–294.
[17] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730* (2018).
[18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013).
[19] Yifan Nie, Yanling Li, and Jian-Yun Nie. 2018. Empirical Study of Multi-level Convolution Models for IR Based on Representations and Interactions. In *ICTIR*. 59–66.
[20] Gustavo Penha, Alexandru Balan, and Claudia Hauff. 2019. Introducing MANtIS: a novel Multi-Domain Information Seeking Dialogues Dataset. *arXiv preprint arXiv:1912.04639* (2019).
[21] Gustavo Penha and Claudia Hauff. 2019. Curriculum Learning Strategies for IR: An Empirical Study on Conversation Response Ranking. *arXiv preprint arXiv:1912.08555* (2019).
[22] Minghui Qiu, Feng-Lin Li, Siyu Wang, Xing Gao, Yan Chen, Weipeng Zhao, Haiqing Chen, Jun Huang, and Wei Chu. 2017. Alime chat: A sequence to sequence and rerank based chatbot engine. In *ACL*. 498–503.
[23] Chen Qu, Liu Yang, W Bruce Croft, Johanne R Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and characterizing user intent in information-seeking conversations. In *SIGIR*. ACM, 989–992.
[24] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. 2009. *Dataset Shift in Machine Learning*. The MIT Press.
[25] Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *EMNLP*. 583–593.
[26] Sebastian Ruder. 2019. *Neural Transfer Learning for Natural Language Processing*. Ph.D. Dissertation. National University of Ireland, Galway.
[27] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for web search. In *WWW*. 373–374.
[28] 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 *IJCAI*. 4382–4388.
[29] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *arXiv preprint arXiv:1511.04108* (2015).
[30] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. 2017. A deeper look at dataset bias. In *Domain adaptation in computer vision applications*. 37–55.
[31] Antonio Torralba, Alexei A Efros, et al. 2011. Unbiased look at dataset bias. In *CVPR*, Vol. 1. 7.
[32] Brandon Tran, Maryam Karimzadehgan, Rama Kumar Pasumarthi, Michael Bendersky, and Donald Metzler. 2019. Domain Adaptation for Enterprise Email Search. In *SIGIR*. 25–34.
[33] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *CVPR*. 7167–7176.
[34] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).
[35] 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 *ACL*, Vol. 1. 496–505.
[36] Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In *SIGIR*. 55–64.
[37] Liu Yang, Qingyao Ai, Damiano Spina, Ruey-Cheng Chen, Liang Pang, W Bruce Croft, Jiafeng Guo, and Falk Scholer. 2016. Beyond factoid QA: effective methods for non-factoid answer sentence retrieval. In *ECIR*. 115–128.
[38] Liu Yang, Junjie Hu, Minghui Qiu, Chen Qu, Jianfeng Gao, W Bruce Croft, Xiaodong Liu, Yelong Shen, and Jingjing Liu. 2019. A Hybrid Retrieval-Generation Neural Conversation Model. *arXiv preprint arXiv:1904.09068* (2019).
[39] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In *SIGIR*. 245–254.