Domain Agnostic Real-Valued Specificity Prediction

Wei-Jen Ko,[*] **Greg Durrett,**[*] **Junyi Jessy Li**[†]
[*]Department of Computer Science
[†]Department of Linguistics
The University of Texas at Austin
wjko@cs.utexas.edu, gdurrett@cs.utexas.edu, jessy@austin.utexas.edu

Abstract

Sentence specificity quantifies the level of detail in a sentence, characterizing the organization of information in discourse. While this information is useful for many downstream applications, specificity prediction systems predict very coarse labels (binary or ternary) and are trained on and tailored toward specific domains (e.g., news). The goal of this work is to generalize specificity prediction to domains where no labeled data is available and output more nuanced real-valued specificity ratings.

We present an unsupervised domain adaptation system for sentence specificity prediction, specifically designed to output real-valued estimates from binary training labels. To calibrate the values of these predictions appropriately, we regularize the posterior distribution of the labels towards a reference distribution. We show that our framework generalizes well to three different domains with 50%-68% mean absolute error reduction than the current state-of-the-art system trained for news sentence specificity. We also demonstrate the potential of our work in improving the quality and informativeness of dialogue generation systems.

1 Introduction

The specificity of a sentence measures its "quality of belonging or relating uniquely to a particular subject"[1] (Lugini and Litman 2017). It is often pragmatically defined as the level of detail in the sentence (Louis and Nenkova 2011a; Li and Nenkova 2015). When communicating, specificity is adjusted to serve the intentions of the writer or speaker (Grice 1975). In the examples below, the second sentence is clearly more specific than the first one:

> **Ex1:** This brand is very popular and many people use its products regularly.
> **Ex2:** Mascara is the most commonly worn cosmetic, and women will spend an average of $4,000 on it in their lifetimes.

Studies have demonstrated the important role of sentence specificity in reading comprehension (Dixon 1987) and in establishing common ground in dialog (Djalali et al. 2011). It has also been shown to be a key property in analyses and applications, such as summarization (Louis and Nenkova 2011b), argumentation mining (Swanson, Ecker, and Walker 2015), political discourse analysis (Cook 2016), student discussion assessment (Luo and Litman 2016; Lugini and Litman 2017), deception detection (Kleinberg et al. 2017), and dialogue generation (Zhang et al. 2018).

Despite their usefulness, prior sentence specificity predictors (Louis and Nenkova 2011a; Li and Nenkova 2015; Lugini and Litman 2017) are trained with sentences from specific domains (news or classroom discussions), and have been found to fall short when applied to other domains (Kleinberg et al. 2017; Lugini and Litman 2017). They are also trained to label a sentence as either general or specific (Louis and Nenkova 2011a; Li and Nenkova 2015), or low/medium/high specificity (Lugini and Litman 2017), though in practice specificity has been analyzed as a continuous value (Louis and Nenkova 2011b; 2013; Swanson, Ecker, and Walker 2015; Cook 2016; Luo and Litman 2016; Kleinberg et al. 2017), as it should be (Li et al. 2016b).

In this work, we present an unsupervised domain adaptation system for sentence specificity prediction, specifically designed to output real-valued estimates. It effectively generalizes sentence specificity analysis to domains where no labeled data is available, and outputs values that are close to the real-world distribution of sentence specificity.

Our main framework is an unsupervised domain adaptation system based on Self-Ensembling (Tarvainen and Valpola 2017; French, Mackiewicz, and Fisher 2018) that simultaneously reduces source prediction errors and generates feature representations that are robust against noise and across domains. Past applications of this technique have focused on computer vision problems; to make it effective for text processing tasks, we modify the network to better utilize labeled data in the source domain and explore several data augmentation methods for text.

We further propose a posterior regularization technique (Ganchev et al. 2010) that generally applies to the scenario where it is easy to get coarse-grained categories of labels, but fine-grained predictions are needed. Specifically, our regularization term seeks to move the distribution of the classifier posterior probabilities closer to that of a prespecified target distribution, which in our case is a specificity distribution derived from the source domain.

Experimental results show that our system generates more

[1] Definition from the Oxford Dictionary

accurate real-valued sentence specificity predictions that correlate well with human judgment, across three domains that are vastly different from the source domain (news): Twitter, Yelp reviews and movie reviews. Compared to a state-of-the-art system trained on news data (Li and Nenkova 2015), our best setting achieves a 50%-68% reduction in mean absolute error and increases Kendall's Tau and Spearman correlations by 0.07-0.10 and 0.12-0.13, respectively.

Finally, we conduct a task-based evaluation that demonstrates the usefulness of sentence specificity prediction in open-domain dialogue generation. Prior work showed that the quality of responses from dialog generation systems can be significantly improved if short examples are removed from training (Li et al. 2017), potentially preventing the system from overly favoring generic responses (Sordoni et al. 2015; Mou et al. 2016). We show that predicted specificity works more effectively than length, and enables the system to generate more diverse and informative responses with better quality.

In sum, the paper's contributions are as follows:

- An unsupervised domain adaptation framework for sentence specificity prediction, available at https://github.com/wjko2/Domain-Agnostic-Sentence-Specificity-Prediction
- A regularization method to derive real-valued predictions from training data with binary labels;
- A task-based evaluation that shows the usefulness of our system in generating better, more informative responses in dialogue.

2 Task setup

With unsupervised domain adaptation, one has access to labeled sentence specificity in one source domain, and unlabeled sentences in all target domains. The goal is to predict the specificity of target domain data. Our source domain is news, the only domain with publicly available labeled data for training (Louis and Nenkova 2011a). We crowdsource sentence specificity for evaluation for three target domains: Twitter, Yelp reviews and movie reviews. The data is described in Section 4.

Existing sentence specificity labels in news are *binary*, i.e., a sentence is either general or specific. However, in practice, real-valued estimates of sentence specificity are widely adopted (Louis and Nenkova 2011b; 2013; Swanson, Ecker, and Walker 2015; Cook 2016; Luo and Litman 2016; Kleinberg et al. 2017). Most of these work directly uses the classifier posterior distributions, although we will later show that such distributions do not follow the true distribution of sentence specificity (see Figure 4, *Speciteller* vs. *real*). We aim to produce accurate real-valued specificity estimates despite the binary training data. Specifically, the test sentences have real-valued labels between 0 and 1. We evaluate our system using mean absolute error, Kendall's Tau and Spearman correlation.

3 Architecture

Our core technique is the Self-Ensembling (Tarvainen and Valpola 2017; French, Mackiewicz, and Fisher 2018) of a single base classification model that utilizes data augmentation, with a distribution regularization term to generate accurate real-valued predictions from binary training labels.

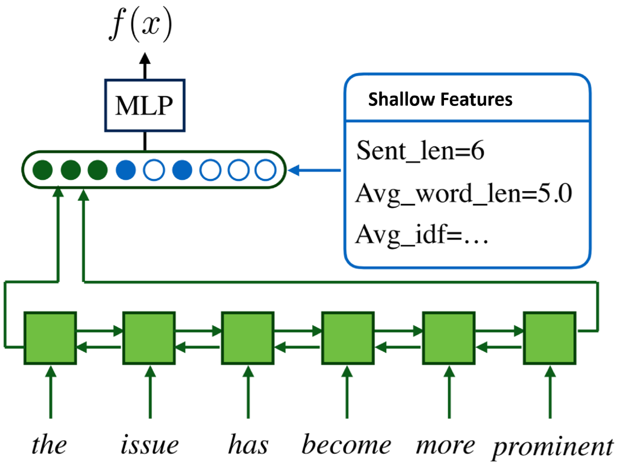

Figure 1: Base model for sentence specificity prediction. The sentence x is encoded with a BiLSTM combined with sparse features, and fed to a MLP to predict specificity $f(x)$.

3.1 Base model

Figure 1 depicts the base model for sentence specificity prediction. The overall structure is similar to that in Lugini and Litman (2017). Each word in the sentence is encoded into an embedding vector and the sentence is passed through a bidirectional Long-Short Term Memory Network (LSTM) (Hochreiter and Schmidhuber 1997) to generate a representation of the sentence. This representation is concatenated with a series of hand crafted features, and then passed through a multilayer perceptron to generate specificity predictions $f(x) \in [0, 1]$. Training treats these as probabilities of the positive class, but we will show in Section 3.3 how they can be adapted to make real-valued predictions.

Our hand crafted features are taken from Li and Nenkova (2015), including: the number of tokens in the sentence; the number of numbers, capital letters and punctuations in the sentence, normalized by sentence length; the average number of characters in each word; the fraction of stop words; the number of words that can serve as explicit discourse connectives; the fraction of words that have sentiment polarity or are strongly subjective; the average familiarity and imageability of words; and the minimum, maximum, and average inverse document frequency (idf) over the words in the sentence.[2] In preliminary experiments, we found that combining hand-crafted features with the BiLSTM performed better than using either one individually.

3.2 Unsupervised domain adaptation

Our unsupervised domain adaptation framework is based on Self-Ensembling (French, Mackiewicz, and Fisher 2018);

[2] We use existing idf values provided by (Li and Nenkova 2015) calculated from the New York Times corpus (Sandhaus 2008).

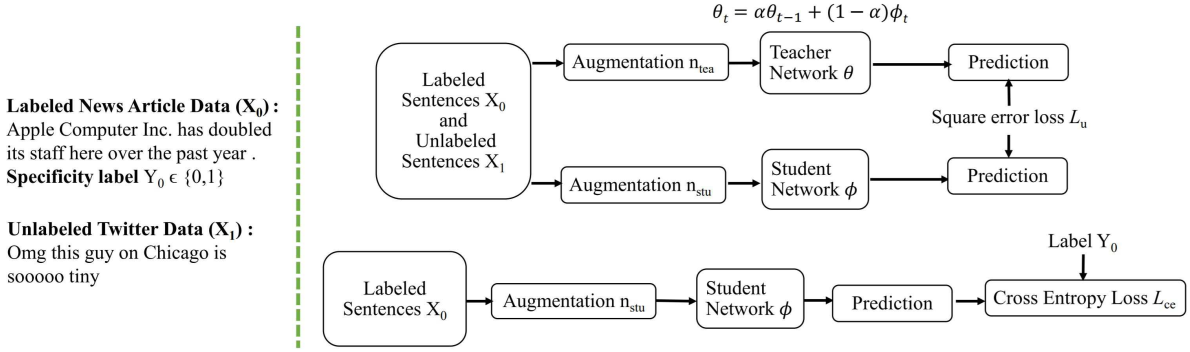
$\theta_t = \alpha\theta_{t-1} + (1-\alpha)\phi_t$
Labeled News Article Data (X_0) :
Apple Computer Inc. has doubled its staff here over the past year .
Specificity label $Y_0 \in \{0,1\}$
Unlabeled Twitter Data (X_1) :
Omg this guy on Chicago is sooooo tiny
Labeled Sentences X_0 and Unlabeled Sentences X_1
Augmentation n_{tea}
Teacher Network θ
Prediction
Square error loss L_u
Augmentation n_{stu}
Student Network ϕ
Prediction
Label Y_0
Labeled Sentences X_0
Augmentation n_{stu}
Student Network ϕ
Prediction
Cross Entropy Loss L_{ce}

Figure 2: Our unsupervised domain adaptation network, showing the consistency component for representation learning (above) and the prediction component (below). We also show examples of a labeled source sentence and an unlabeled target sentence.

we lay out the core ideas first and then discuss modifications to make the framework suitable for textual data.

There are two ideas acting as the driving force behind Self-Ensembling. First, adding noise to each data point x can help regularize the model by encouraging the model prediction to stay the same regardless of the noise, creating a *manifold* around the data point where predictions are invariant (Rasmus et al. 2015). This can be achieved by minimizing a *consistency loss* L_u between the predictions. Second, temporal ensembling—ensemble the same model trained at different time steps—is shown to be beneficial to prediction especially in semi-supervised cases (Laine and Aila 2017); in particular, we average model parameters from each time step (Tarvainen and Valpola 2017).

These two ideas are realized by using a student network and a teacher network. The parameters of the teacher network are the exponential average of that of the student network, making the teacher a temporal ensemble of the student. Distinct noise augmentations are applied to the input of each network, hence the consistency loss L_u is applied between student and teacher predictions. The student learns from labeled source data and minimizes the supervised cross-entropy loss L_{ce}. Domain adaptation is achieved by minimizing the consistency loss between the two networks, which can be done with unlabeled target data. The overall loss function is the weighted sum of L_{ce} and L_u. Figure 2 depicts the process.

Concretely, the student and teacher networks are of identical *structure* following the base model (Section 3.1), but features distinct *noise augmentation*. The student network learns to predict sentence specificity from labeled source domain data. The input sentences are augmented with noise n_{stu}. The teacher network predicts the specificity of each sentence with a different noise augmentation n_{tea}. Parameters of the teacher network θ are updated each time step to be the exponential moving average of the corresponding parameters in the student network. The teacher parameter θ_t at each time step t is

$$\theta_t = \alpha\theta_{t-1} + (1-\alpha)\phi_t \tag{1}$$

where α is the degree of weighting decay, a constant between 0 and 1. ϕ denotes parameters for the student network.

The consistency loss is defined as the squared difference between the predictions of the student and teacher networks (Tarvainen and Valpola 2017):

$$L_u = (f(n_{stu}(x)|\phi) - f(n_{tea}(x)|\theta))^2 \tag{2}$$

where f denotes the base network and x denotes the input sentence. The teacher network is not involved when minimizing the supervised loss L_{ce}.

An important difference between our work and French, Mackiewicz, and Fisher (2018) is that in their work only unlabeled target data contributes to the consistency loss L_u. Instead, we use both source and target sentences, bringing the predictions of the two networks close to each other not only on the target domain, but also the source domain. Unlike many vision tasks where predictions intuitively stay the same with different types of image augmentation (e.g., transformation and scaling), text is more sensitive to noise. Self-Ensembling relies on heavy augmentation on both student and teacher networks, and our experiments revealed that incorporating source data in the consistency loss term mitigates additional biases from noise augmentation.

At training time, the teacher network's parameters are fixed during gradient descent, and the gradient only propagates through the student network. After each update of the student network, we recalculate the weights of the teacher network using exponential moving average. At testing time, we use the teacher network for prediction.

Noise augmentation An important factor contributing to the effectiveness of Self-Ensembling is applying noise to the inputs to make them more robust against domain shifts. For computer vision tasks, augmentation techniques including affine transformation, scaling, flipping and cropping could be used (French, Mackiewicz, and Fisher 2018). However these operations could not be used on text. We designed several noise augmentations for sentences, including: adding Gaussian noise to both the word embeddings and shallow features; randomly removing words in a sentence; substituting word embeddings with a random vector, or a zero vector like applying dropout. To produce enough variations of data, augmentation is applied to ∼half of the words in a sentence.

3.3 Regularizing the posterior distribution

Sentence specificity prediction is a task where the existing training data have binary labels, while real-valued outputs are desirable. Prior work has directly used classifier posterior probabilities. However, the posterior distribution and the true specificity distribution are quite different (see Figure 4, *Speciteller* vs. *real*). We propose a regularization term to bridge the gap between the two.

Specifically, we view the posterior probability distribution as a latent distribution, which allows us to apply a variant of posterior regularization (Ganchev et al. 2010), previously used to apply pre-specified constraints to latent variables in structured prediction. Here, we apply a distance penalty between the latent distribution and a pre-specified reference distribution (which, in our work, is from the source domain). Li et al. (2016b) found that in news, the distribution of sentence specificity is bell shaped, similar to that of a Gaussian. Our analysis on sentence specificity for three target domains yields the same insights (Figure 3). We explored two regularization formulations, with and without assuming that the two distributions are Gaussian. Both were successful and achieved similar performance.

Let μ_p and σ_p be the mean and standard deviation of the predictions (posterior probabilities) in a batch. The first formulation assumes that the predictions and reference distributions are Gaussian. It uses the KL divergence between the predicted distribution $p(x) = N(\mu_p, \sigma_p)$ and the reference Gaussian distribution $r(x) = N(\mu_r, \sigma_r)$. The distribution regularization loss can be written as:

$$L_d = KL(r||p) = \log \frac{\sigma_p}{\sigma_r} + \frac{\sigma_r^2 + (\mu_r - \mu_p)^2}{2\sigma_p^2} - \frac{1}{2} \quad (3)$$

The second formulation does not assume Gaussian distributions and only compares the mean and standard deviation of the two distributions using a weighting term β:

$$L_d = |\sigma_r - \sigma_p| + \beta|\mu_r - \mu_p|. \quad (4)$$

Combining the regularization term L_d into a single objective, the total loss is thus:

$$L = L_{ce} + c_1 L_u + c_2 L_d \quad (5)$$

where L_{ce} is the cross entropy loss for the source domain predictions, L_u is the consistency loss. c_1 and c_2 are weighting hyperparameters.

In practice, this regularization term serves a second purpose. After adding the consistency loss L_u, we observed that the predictions are mostly close to each other with values between 0.4 and 0.6, and their distribution resembles a Gaussian with very small variance (c.f. Figure 4 line *SE+A*). This might be due to the consistency loss pulling all the predictions together, since when all predictions are identical, the loss term will be zero. This regularization can be used to counter this effect, and avoid the condensation of predicted values. Finally, this regularization is distinct from class imbalance loss terms such as that used in French, Mackiewicz, and Fisher (2018), which we found early on to hurt performance rather than helping.

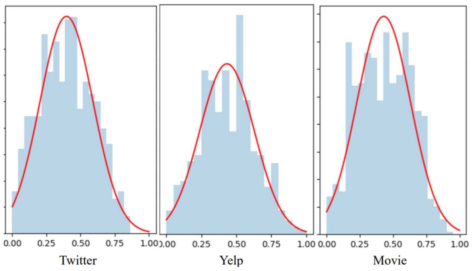

Figure 3: Histograms of specificity distribution for each target domain, shown with a fitted Gaussian distribution.

4 Datasets for sentence specificity

4.1 Source domain

The source domain for sentence specificity is news, for which we use three publicly available labeled datasets: (1) training sentences from Louis and Nenkova (2011a) and Li and Nenkova (2015), which consists of 1.4K general and 1.4K specific sentences from the Wall Street Journal. (2) 900 news sentences crowdsourced for binary general/specific labels (Louis and Nenkova 2012); 55% of them are specific. (3) 543 news sentences from Li et al. (2016b). These sentences are rated on a scale of $0 - 6$, so for consistency with the rest of the training labels, we pick sentences with average rating > 3.5 as general examples and those with average rating < 2.5 as specific. In total, we have 4.3K sentences with binary labels in the source domain.

4.2 Target domains

We evaluate on three target domains: Twitter, Yelp and movie reviews. Since no annotated data exist for these domains, we crowdsource specificity for sentences sampled from each domain using Amazon Mechanical Turk.

We follow the context-independent annotation instructions from Li et al. (2016b). Initially, 9 workers labeled specificity for 1000 sentences in each domain on a scale of 1 (very general) to 5 (very specific), which we rescaled to 0, 0.25, 0.5, 0.75, and 1. Inter-annotator agreement (IAA) is calculated using average Cronbach's alpha (Cronbach 1951) values for each worker. For quality control, we exclude workers with IAA below 0.3, and include the remaining sentences that have at least 5 raters. Our final IAA values fall between 0.68-0.70, compatible with the reported 0.72 from expert annotators in Li et al. (2016b). The final specificity value is aggregated to be the average of the rescaled ratings. We also use large sets of unlabeled data in each domain:

- Twitter: 984 tweets annotated, 50K unlabeled, sampled from Preoţiuc-Pietro et al. (2017).
- Yelp: 845 sentences annotated, 95K unlabeled, sampled from the Yelp Dataset Challenge 2015 (Zhang, Zhao, and LeCun 2015).

domain	mean	std.dev
Twitter	0.405	0.193
Yelp	0.419	0.198
Movie	0.426	0.206
News (Li et al. 2016b)	0.417	0.227

Table 1: Mean and standard deviation of sentence specificity for each domain.

- Movie: 920 sentences annotated, 12K unlabeled, sampled from Pang and Lee (2004).

Figure 3 shows the distribution of ratings for the annotated data. We also plot a fitted Gaussian distribution for comparison. Clearly, most sentences have mid-range specificity values, consistent with news sentences (Li et al. 2016b). Interestingly the mean and variance for the three distributions are similar to each other and to those from Li et al. (2016b), as show in Table 1. Therefore, we use the source distribution (news, Li et al. (2016b)) as the reference distribution for posterior distribution regularization, and set μ_r, σ_r to 0.417 and 0.227 accordingly.

5 Experiments

We now evaluate our framework on predicting sentence specificity for the three target domains. We report experimental results on a series of settings to evaluate the performance of different components.

5.1 Systems

Length baseline This simple baseline predicts specificity proportionally to the number of words in the sentence. Shorter sentences are predicted as more general and longer sentences are predicted as more specific.

Speciteller baseline Speciteller (Li and Nenkova 2015) is a semi-supervised system trained on news data with binary labels. The posterior probabilities of the classifier are used directly as specificity values.

Self-ensembling baseline (SE) Our system with the teacher network, but only using exponential moving average (without the consistency loss or distribution regularization).

Distribution only (SE+D) Our system with distribution regularization L_d using the mean and standard deviation (Eq. 4), but without the consistency loss L_u.

Adaptation only (SE+A) Our system with the consistency loss L_u, but without distribution regularization L_d.

SE+AD (KL) Our system with both L_u and L_d using KL divergence (Eq. 3).

SE+AD (mean-std) Our system with both L_u and L_d using mean and standard deviation (Eq. 4).

SE+AD (no augmentation) We also show the importance of noise augmentation, by benchmarking the same setup as SE+AD (mean-std) without data augmentation.

5.2 Training details

Hyperparameters are tuned on a validation set of 200 tweets that doesn't overlap with the test set. We then use this set of parameters for all testing domains. The LSTM encoder generates 100-dimensional representations. For the multilayer perceptron, we use 3 fully connected 100-dimensional layers. We use ReLU activation with batch normalization. For the Gaussian noise in data augmentation, we use standard deviation 0.1 for word embeddings and 0.2 for shallow features. The probabilities of deleting a word and replacing a word vector are 0.15. The exponential moving average decay α is 0.999. Dropout rate is 0.5 for all layers. The batch size is 32. $c_1 = 1000$, $c_2 = 10$ for KL loss and 100 for mean and std.dev loss. $\beta = 1$. We fix the number of training to be 30 epochs for SE+A and SE+AD, 10 epochs for SE, and 15 epochs for SE+D. We use the Adam optimizer with learning rate 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.999$. As discussed in Section 4, posterior distribution regularization parameters μ_r, σ_r are set to be those from Li et al. (2016b).

5.3 Evaluation metrics

We use 3 metrics to evaluate real-valued predictions: (1) the **Spearman correlation** between the labeled and predicted specificity values, higher is better; (2) the pairwise **Kendall's Tau** correlation, higher is better; (3) **mean absolute error (MAE)**: $\sum |Y - X|/n$, lower is better.

5.4 Results and analysis

Table 2 shows the full results for the baselines and each configuration of our framework. For analysis, we also plot in Figure 4 the true specificity distributions in Twitter test set, predicted distributions for Speciteller, the Self-Ensembling baseline (SE), SE with adaptation (SE+A) and also with distribution regularization (SE+AD).

Speciteller, which is trained on news sentences, cannot generalize well to other domains, as it performs worse than just using sentence length for two of the three domains (Yelp and Movie). From Figure 4, we can see that the prediction mass of Speciteller is near the extrema values 0 and 1, and the rest of the predictions falls uniformly in between. These findings confirm the need of a generalizable system.

Across all domains and all metrics, the best performing system is our full system with domain adaptation and distribution regularization (SE+AD with mean-std or KL), showing that the system generalizes well across different domains. Using a paired Wilcoxon test, it significantly ($p < 0.001$) outperforms Speciteller in terms of MAE; it also achieved higher Spearman and Kendall's Tau correlations than both Length and Speciteller.

Component-wise, the Self-Ensembling baseline (SE) achieves significantly lower MAE than Speciteller, and higher correlations than either baseline. Figure 4 shows that unlike Speciteller, the SE baseline does not have most of its prediction mass near 0 and 1, demonstrating the effectiveness of temporal ensembling. Using both the consistency loss L_u and the distribution regularization L_d achieves the best results on all three domains; however adding only L_u (SE+A) or L_d (SE+D) improves for some measures or domains but not all. This shows that both terms are crucial to make the system robust across domains.

The improvements from distribution regularization are visualized in Figure 4. With SE+A, most of the predicted la-

	Baselines					SE+AD		
metric	Length	Speciteller	SE	SE+D	SE+A	mean-std	KL	no aug.
				Twitter				
Spearman	0.445	0.553	0.622±0.028	0.610±0.011	0.670±0.004	0.676±0.004	**0.679±0.005**	0.666
Kendall's Tau	0.324	0.413	0.437±0.024	0.427±0.008	0.480±0.004	**0.487±0.005**	0.482±0.006	0.480
MAE	-	0.237	0.148±0.006	0.125±0.001	0.151±0.005	**0.113±0.001**	0.115±0.002	0.115
				Yelp				
Spearman	0.676	0.633	0.731±0.009	0.721±0.009	0.735±0.001	**0.750±0.016**	0.743±0.010	0.728
Kendall's Tau	0.522	0.481	0.548±0.006	0.536±0.008	0.544±0.001	**0.555±0.010**	0.546±0.014	0.533
MAE	-	0.325	0.165±0.016	0.120±0.010	0.137±0.005	0.107±0.003	**0.105±0.002**	0.109
				Movie				
Spearman	0.581	0.575	0.684±0.004	0.664±0.004	0.680±0.007	0.702±0.003	**0.706±0.030**	0.669
Kendall's Tau	0.435	0.418	0.498±0.004	0.487±0.010	0.502±0.010	0.519±0.015	**0.522±0.024**	0.484
MAE	-	0.226	0.143±0.009	0.124±0.009	0.148±0.001	**0.114±0.001**	**0.114±0.006**	0.118

Table 2: Sentence specificity prediction results (across 3 runs) for: Length and Speciteller baselines, Self-Ensembling baseline (SE), SE with mean-std distribution regularization (SE+D), with consistency loss (SE+A), and both (SE+AD). Also showing SE+AD without data augmentation (no aug.).

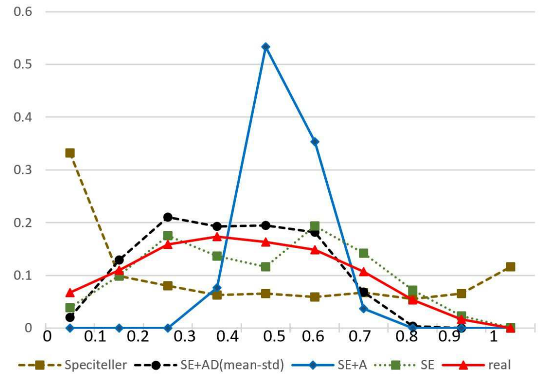

Figure 4: Distribution of predictions for: Speciteller, Self-Ensembling baseline (SE), SE with consistency loss (SE+A), and also with distribution regularization (SE+AD).

bels are between 0.4 and 0.6. Applying distribution regularization (SE+AD) makes them much closer to the real distribution of specificity. With respect to the two formulations of regularization (KL and mean-std), both are effective in generating more accurate real-valued estimates. Their performances are comparable, hence using only the mean and standard deviation values, without explicitly modeling the reference Gaussian distribution, works equally well.

Finally, without data augmentation (column no aug.), the correlations are clearly lower than our full model, stressing the importance of data augmentation in our framework.

6 Specificity in dialogue generation

We also evaluate our framework in open-domain dialogue. This experiment also presents a case for the usefulness of an effective sentence specificity system in dialogue generation.

With SEQ2SEQ dialogue generation models, (Li et al. 2017) observed significant quality improvement by removing training examples with short responses during preprocessing; this is potentially related to this type of models favor generating non-informative, generic responses (Sordoni et al. 2015; Mou et al. 2016). We show that filtering training data by predicted specificity results in responses of higher quality and informativeness than filtering by length.

6.1 Task and settings

We implemented a SEQ2SEQ question-answering bot with attention using OpenNMT(Klein et al. 2017). The bot is trained on OpenSubtitles (Tiedemann 2009) following prior work (Li et al. 2017). We restrict the instances to question-answer pairs by selecting consecutive sentences with the first sentence ending with question mark, the second sentence without question mark and follows the first sentence by less than 20 seconds, resulting in a 14M subcorpus.

The model uses two hidden layers of size 2048, optimized with Adam with learning rate 0.001. The batch size is 64. While decoding, we use beam size 5; we block repeating n-grams, and constrain the minimum prediction length to be 5. These parameters are tuned on a development set. We compare two ways to filter training data during preprocessing:

Remove Short: Following Li et al. (2017), remove training examples with length of responses shorter than a threshold of 5. About half of the data are removed.

Remove General: Remove predicted general responses from training examples using our system. We use the responses in the OpenSubtitles training set as the unlabeled target domain data during training. We remove the least specific responses such that the resulting number of examples is the same as Remove Short.

For fair comparison, at test time, we adjust the length penalty described in Wu et al. (2016) for both models, so the average response lengths are the same among both models.

	Remove Short	Remove General
unigram diversity	0.0177	**0.0199**
bigram diversity	0.0833	**0.0977**
perplexity	51.97	**46.92**

Table 3: Perplexity and diversity scores for Remove Short and Remove General.

	Remove Short wins	Remove General wins	tie
informativeness	29.0	38.9	32.1
quality	28.9	34.7	36.4

Table 4: Human evaluation results for Remove Short and Remove General.

6.2 Evaluation

We use automatic measures and human evaluation as in Li et al. (2016a) and Li et al. (2017).

Table 3 shows the diversity and perplexity of the responses. Diversity is calculated as the type-token ratio of unigrams and bigrams. The test set for these two metrics is a random sample of 10K instances from OpenSubtitles that doesn't overlap with the training set. Clearly, filtering training data according to specificity results in more diverse responses with lower perplexity than filtering by length.

We also crowdsource human evaluation for quality; in addition, we evaluate the systems for response informativeness. Note that in our instructions, informativeness means the *usefulness* of information and is a distinct measure from specificity. The original training data of specificity are linguistically annotated and involves only the change in the level of details (Louis and Nenkova 2011a). Separate experiments are conducted to avoid priming. We use a test set of 388 instances, including questions randomly sampled from OpenSubtitles that doesn't overlap with the training set, and 188 example questions from previous dialogue generation papers, including Vinyals and Le (2015). We use Amazon MechenicalTurk for crowdsourcing. 7 workers chose between the two responses to the same question.

Table 4 shows the human evaluation comparing Remove Short vs. Remove General. Removing predicted general responses performs better than removing short sentences, on both informativeness and quality and on both test sets. This shows that sentence specificity is a superior measure for training data preprocessing than sentence length.

7 Related Work

Sentence specificity prediction as a task is proposed by Louis and Nenkova (2011a), who repurposed discourse relation annotations from WSJ articles (Prasad et al. 2008) for sentence specificity training. Li and Nenkova (2015) incorporated more news sentences as unlabeled data. Lugini and Litman (2017) developed a system to predict sentence specificity for classroom discussions, however the data is not publicly available. All these systems are classifiers trained with categorical data (2 or 3 classes).

We use Self-Ensembling (French, Mackiewicz, and Fisher 2018) as our underlying framework. Self-Ensembling builds on top of Temporal Ensembling (Laine and Aila 2017) and the Mean-Teacher network (Tarvainen and Valpola 2017), both of which originally proposed for semi-supervised learning. In visual domain adaptation, Self-Ensembling shows superior performance than many recently proposed approaches (Ganin and Lempitsky 2015; Ghifary et al. 2016; Russo et al. 2018; Haeusser et al. 2017; Tzeng et al. 2017; Sankaranarayanan et al. 2018), including GAN-based approaches. To the best of our knowledge, this approach has not been used on language data.

8 Conclusion

We present a new model for predicting sentence specificity. We augment the Self-Ensembling method (French, Mackiewicz, and Fisher 2018) for unsupervised domain adaptation on text data. We also regularize the distribution of predictions to match a reference distribution. Using only binary labeled sentences from news articles as source domain, our system could generate real-valued specificity predictions on different target domains, significantly outperforming previous work on sentence specificity prediction. Finally, we show that sentence specificity prediction can potentially be beneficial in improving the quality and informativeness of dialogue generation systems.

Acknowledgments

This research was partly supported by the Amazon Alexa Graduate Fellowship. We thank the anonymous reviewers for their helpful comments.

References

Cook, I. P. 2016. *Content and Context: Three Essays on Information in Politics*. Ph.D. Dissertation, University of Pittsburgh.

Cronbach, L. J. 1951. Coefficient alpha and the internal structure of tests. *Psychometrika* 16(3):297–334.

Dixon, P. 1987. The processing of organizational and component step information in written directions. *Journal of memory and language* 26(1):24.

Djalali, A.; Clausen, D.; Lauer, S.; Schultz, K.; and Potts, C. 2011. Modeling expert effects and common ground using questions under discussion. In *AAAI Fall Symposium: Building Representations of Common Ground with Intelligent Agents*.

French, G.; Mackiewicz, M.; and Fisher, M. 2018. Self-ensembling for visual domain adaptation. In *ICLR*.

Ganchev, K.; Graça, J.; Gillenwater, J.; and Taskar, B. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research* 11:2001–2049.

Ganin, Y., and Lempitsky, V. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*.

Ghifary, M.; Kleijn, W. B.; Zhang, M.; Balduzzi, D.; and Li, W. 2016. Deep reconstruction-classification networks for unsupervised domain adaptation. In *ECCV*.

Grice, H. P. 1975. Logic and conversation. *Syntax and Semantics* 3:41–58.

Haeusser, P.; Frerix, T.; Mordvintsev, A.; and Cremers, D. 2017. Associative domain adaptation. In *ICCV*.

Hochreiter, S., and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation* 9(8):1735–1780.

Klein, G.; Kim, Y.; Deng, Y.; Senellart, J.; and Rush, A. M. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *ACL*.

Kleinberg, B.; Mozes, M.; Arntz, A.; and Verschuere, B. 2017. Using named entities for computer-automated verbal deception detection. *Journal of forensic sciences* 63(3):714–723.

Laine, S., and Aila, T. 2017. Temporal ensembling for semi-supervised learning. In *ICLR*.

Li, J. J., and Nenkova, A. 2015. Fast and accurate prediction of sentence specificity. In *AAAI*.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2016a. A diversity-promoting objective function for neural conversation models. In *NAACL*.

Li, J. J.; O'Daniel, B.; Wu, Y.; Zhao, W.; and Nenkova, A. 2016b. Improving the annotation of sentence specificity. In *LREC*.

Li, J.; Monroe, W.; Shi, T.; Jean, S.; Ritter, A.; and Jurafsky, D. 2017. Adversarial learning for neural dialogue generation. In *EMNLP*.

Louis, A., and Nenkova, A. 2011a. Automatic identification of general and specific sentences by leveraging discourse annotations. In *IJCNLP*.

Louis, A., and Nenkova, A. 2011b. Text specificity and impact on quality of news summaries. In *Workshop on Monolingual Text-To-Text Generation*.

Louis, A., and Nenkova, A. 2012. A corpus of general and specific sentences from news. In *LREC*.

Louis, A., and Nenkova, A. 2013. A corpus of science journalism for analyzing writing quality. *Dialogue & Discourse* 4(2):87–117.

Lugini, L., and Litman, D. 2017. Predicting specificity in classroom discussion. In *Workshop on Innovative Use of NLP for Building Educational Applications*.

Luo, W., and Litman, D. 2016. Determining the quality of a student reflective response. In *Florida Artificial Intelligence Research Society Conference*.

Mou, L.; Song, Y.; Yan, R.; Li, G.; Zhang, L.; and Jin, Z. 2016. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In *COLING*.

Pang, B., and Lee, L. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *ACL*.

Prasad, R.; Dinesh, N.; Lee, A.; Miltsakaki, E.; Robaldo, L.; Joshi, A.; and Webber, B. 2008. The Penn Discourse Treebank 2.0. In *LREC*.

Preoţiuc-Pietro, D.; Liu, Y.; Hopkins, D.; and Ungar, L. 2017. Beyond binary labels: political ideology prediction of twitter users. In *ACL*.

Rasmus, A.; Berglund, M.; Honkala, M.; Valpola, H.; and Raiko, T. 2015. Semi-supervised learning with ladder networks. In *NIPS*.

Russo, P.; Carlucci, F. M.; Tommasi, T.; and Caputo, B. 2018. From source to target and back: symmetric bi-directional adaptive GAN. In *CVPR*.

Sandhaus, E. 2008. The New York Times Annotated Corpus LDC2008T19. *Linguistic Data Consortium*.

Sankaranarayanan, S.; Balaji, Y.; Castillo, C. D.; and Chellappa, R. 2018. Generate to adapt: Aligning domains using generative adversarial networks. In *CVPR*.

Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.-Y.; Gao, J.; and Dolan, B. 2015. A neural network approach to context-sensitive generation of conversational responses. In *NAACL*.

Swanson, R.; Ecker, B.; and Walker, M. 2015. Argument mining: Extracting arguments from online dialogue. In *SIGDIAL*.

Tarvainen, A., and Valpola, H. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*.

Tiedemann, J. 2009. News from OPUS - a collection of multilingual parallel corpora with tools and interfaces. In *RANLP*.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *CVPR*.

Vinyals, O., and Le, Q. 2015. A neural conversational model. In *ICML Deep Learning Workshop*.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; Klingner, J.; Shah, A.; Johnson, M.; Liu, X.; Kaiser, L.; Gouws, S.; Kato, Y.; Kudo, T.; Kazawa, H.; Stevens, K.; Kurian, G.; Patil, N.; Wang, W.; Young, C.; Smith, J.; Riesa, J.; Rudnick, A.; Vinyals, O.; Corrado, G.; Hughes, M.; and Dean, J. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. In *CoRR, abs/1609.08144*.

Zhang, R.; Guo, J.; Fan, Y.; Lan, Y.; Xu, J.; and Cheng, X. 2018. Learning to control the specificity in neural response generation. In *ACL*.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level convolutional networks for text classification. In *NIPS*.