LEATHER: A Framework for Learning to Generate Human-like Text in Dialogue

Anthony Sicilia[1] and **Malihe Alikhani**[1,2]
{anthonysicilia, malihe}@pitt.edu
[1]Intelligent Systems Program and [2]Computer Science Department
University of Pittsburgh, Pittsburgh, PA, USA

Abstract

Algorithms for text-generation in dialogue can be misguided. For example, in task-oriented settings, reinforcement learning that optimizes only task-success can lead to abysmal lexical diversity. We hypothesize this is due to poor theoretical understanding of the objectives in text-generation and their relation to the learning process (i.e., model training). To this end, we propose a new theoretical framework for learning to generate text in dialogue. Compared to existing theories of learning, our framework allows for analysis of the multi-faceted goals inherent to text-generation. We use our framework to develop theoretical guarantees for learners that adapt to unseen data. As an example, we apply our theory to study data-shift within a cooperative learning algorithm proposed for the *GuessWhat?!* visual dialogue game. From this insight, we propose a new algorithm, and empirically, we demonstrate our proposal improves both task-success and human-likeness of the generated text. Finally, we show statistics from our theory are empirically predictive of multiple qualities of the generated dialogue, suggesting our theory is useful for model-selection when human evaluations are not available.

1 Introduction

Generating coherent, human-like text for dialogue remains a challenge. Yet, it is an inseparable component of open domain and task oriented dialogue systems like Alexa and Siri. Undoubtedly, it is also a complex process to learn. Generation based on classification (e.g., next-word prediction) over-emphasizes the likelihood of text, leading to bland qualities, which are not human-like (Holtzman et al., 2019). Meanwhile, framing dialogue generation as a Markov decision process is highly data-inefficient when compared to classification (Kakade, 2003). Further, without careful design of rewards, models can suffer from mode-collapse in dialogue, producing repetitive behaviors that are not human-like (Shekhar et al., 2019). Even carefully designed rule-based systems are brittle in the presence of unforeseen data-shift.

Theoretical analyses of learning are imperative as they provide solutions to these issues. For example, traditional (PAC) learning theory (Valiant, 1984) studies similar issues arising from computational algorithms for learning to classify. Progress in our understanding has been impressive, ranging from comprehensive guarantees on data-efficiency (Shalev-Shwartz and Ben-David, 2014) to insights for algorithm-design when the learner is faced with data-shift (Zhao et al., 2019; Zhang et al., 2019b; Tachet des Combes et al., 2020). While traditional theory may be applicable to simple generation objectives like next-word prediction, it is unfortunately unable to model more diverse goals. That is to say, it is insufficient to study replication of the diverse qualities inherent to human dialogue.

The goal of this paper is to provide a new theory for analyzing the multi-faceted objectives in computational learning of dialogue generation. In particular, we propose LEATHER[1] based on existing theories of computational learning. We demonstrate the utility of LEATHER with a focus on understanding data-shift in learning algorithms. We also show empirical results for a task-oriented visual dialogue game. In detail, we contribute as follows:

1. In Section 3, we propose LEATHER, our novel theory for computational learning of dialogue generation. We use the *GuessWhat?!* visual dialogue game (De Vries et al., 2017) as an example to ground abstract terminology in practice. We conclude Section 3 by applying our theory to analyze a cooperative learning algorithm for *GuessWhat?!*. Our theory unveils harmful shifts in data-distribution that occur during training.
2. In Section 4, we use LEATHER to study the general problem of data-shift in text-generation. We provide new theoretical study that characterizes

[1]**LEA**rning **T**heory for **H**uman-like dialogue gen**ER**ation

Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022, pages 30–53
November 20–23, 2022.

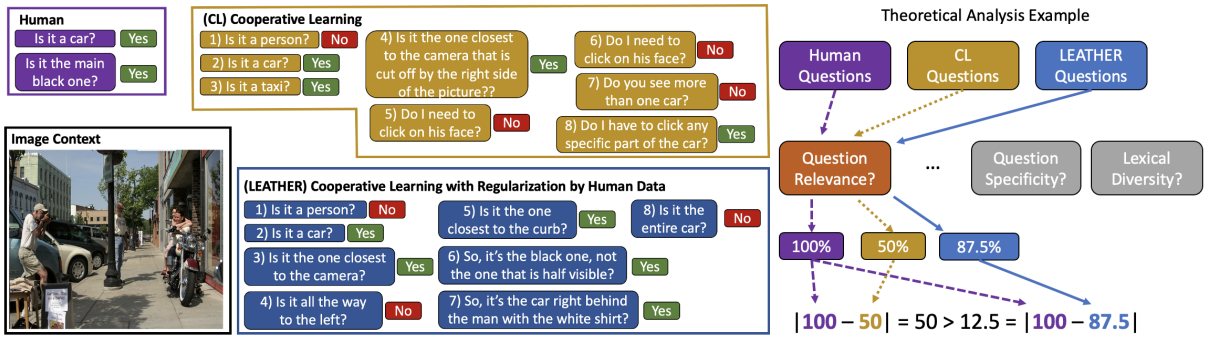

Figure 1: Examples of human and generated dialogue with original cooperative learning algorithm CL (Shekhar et al., 2019) and our learning algorithm motivated by our theory (LEATHER). Roughly, LEATHER works by applying a series of tests to generated dialogue and comparing the test results across the human and generated dialogue. Well-generated dialogue is expected to perform similarly to human dialogue on these tests. The example tests the % of relevant questions. Compared to CL, LEATHER asks more relevant questions and therefore behaves more human-like. Aggregate empirical results in Section 5 echo this trend.

statistical energy as an effective empirical tool for quantifying the impact of data-shift. Aptly, to conclude Section 4, we use energy to motivate an improved learning algorithm for our running example – the *GuessWhat?!* game.

3. In Section 5, empirically, we demonstrate the benefits of our LEATHER-inspired algorithm compared to common baselines. Importantly, we also show our proposed statistic (energy) is predictive of the quality of generated dialogue; i.e., we exhibit a linear relationship. This suggests LEATHER is useful, not only as a theoretical tool for algorithm design, but also as an empirical tool for model-selection.

Our framework is publicly available through experimental code and a Python package.[2]

2 Related Works

Theories of Learning to Generate Text Most widely, text-generation is framed as a classification problem, in which a model predicts the next word provided existing context (e.g., previous words). While common PAC learning analyses do apply to classification, this theory only describes the learner's ability at the next-word prediction task. In some specific cases, instead, PAC analysis has also been used to analyze high-level objectives and motivate conversational strategies (Sicilia et al., 2022b), but this analysis is problem-dependent. In contrast, our work offers a general problem-independent formalism for studying high-level qualities of generated text. Another frequent formalism comes from partially observable Markov decision processes (POMDPs) used to motivate reinforcement learning. For example, see Strub et al. (2017). While POMDPs remedy the issues of typical PAC analysis by supporting implementation of high-level objectives, as we are aware, there are no empirically verified theoretical studies of learning under data-shift in POMDPs. In contrast, we demonstrate LEATHER admits such a theory of learning, using it to predict the human-likeness of generated text under data-shift (where POMDPs fall short).

Theories of Learning with Data-Shift Early learning theoretic models of data-shift in classification and regression are due to Ben-David et al. (2010a,b) and Mansour et al. (2009). While modern approaches are generally similar in spirit, new statistics incorporate increasing information about the learning algorithm (Lipton et al., 2018; Kuroki et al., 2019; Germain et al., 2020; Sicilia et al., 2022a). Ultimately, such techniques tend to improve the predictive capabilities of a theory in practical application (Rabanser et al., 2019; Atwell et al., 2022). Diverse additional approaches to describing the impact of data-shift have also been proposed, for example, using integral probability metrics (Redko et al., 2017, 2020; Shen et al., 2018; Johansson et al., 2019). Unfortunately, existing works focus on classification and regression which, as discussed, are not directly applicable to dialogue generation. Further, this theory does not easily extend to generation (see Section 3.3). Ultimately, using LEATHER, our work derives a new statistic (energy) for predicting changes in model performance, which *is* applicable to dialogue generation.

Evaluation of Generated Text There are many automated metrics for evaluation of generated text including metrics based on n-grams such as BLEU

[2] github.com/anthonysicilia/LEATHER-AACL2022

(Papineni et al., 2002), ROUGE (Lin, 2004), and CIDEr (Vedantam et al., 2015). Automated metrics based on neural models are also becoming more prevalent including BLEURT (Sellam et al., 2020), BertScore (Zhang et al., 2019a), and COSMic (Inan et al., 2021). Bruni and Fernandez (2017) propose use of an adversary to discriminate between human and generated text, evaluating based on the generator's ability to fool the adversary. Human annotation and evaluation, of course, remains the gold-standard. Notably, our proposed framework encapsulates these techniques, since it is suitable for analyzing the impact of the learning process on *all of these evaluation strategies and more* (see Section 3 for examples).

3 Theory with Examples

In this section, we develop our new theoretical framework. To assist our exposition, we use the *GuessWhat?!* visual dialogue game – a variant of the child's game *I Spy* – as a running example. We first describe the game along with our modeling interests within the game. We continue with a description of our theory and then apply this theory to analyze an algorithm that learns to play the game.

3.1 *GuessWhat?!* Visual Dialogue Game

An image and **goal-object** within the image are both randomly chosen. A **question-player** with access to the image asks yes/no questions to an **answer-player** who has access to both the image and goal-object. The question-player's goal is to identify the goal-object. The answer-player's goal is to reveal the goal-object to the question-player by answering the yes/no questions appropriately. The question- and answer-player converse until the question-player is ready to make a guess or at most m questions have been asked.[3] The question-player then guesses which object was the secret goal.

Notation for Human Games To discuss this game within our theoretical framework next, we provide some notation. We assume the possible questions, answers, and objects are respectively confined to the sets $\mathcal{Q}$, $\mathcal{A}$, and $\mathcal{O}$. We also assume a set of possible images $\mathcal{I}$. A game between two human players can be represented by a series of random variables. The image-object pair is represented by the random tuple (I, O). The dialogue between the question- and answer-player is represented by the random-tuple $D = (Q_1, A_1, \ldots, Q_P, A_P)$ with some random length $P \leq m$. Each Q_i is a random question taking value from the set $\mathcal{Q}$ and each A_i is a random answer from the set $\mathcal{A}$.

Notation for Modeled Games From a modeling perspective, in this paper, we focus on the question-player and assume a human answer-player. We consider learning a model that generates the random dialogue $\hat{D} = (\hat{Q}_1, \tilde{A}_1, \ldots \hat{Q}_m, \tilde{A}_m)$ along with a predicted goal object $\hat{O}$.[4] For example, consider the model of Shekhar et al. (2019) we study later. It generates dialogue/predicted goal as below:

$$\begin{aligned} \hat{O} &= \texttt{Gues}_\alpha(\texttt{Enc}_\beta(I, \hat{D})) \\ \hat{Q}_{i+1} &= \texttt{QGen}_\theta(\texttt{Enc}_\beta(I, \hat{Q}_1, \tilde{A}_1, \ldots \hat{Q}_i, \tilde{A}_i) \end{aligned} \tag{1}$$

where, aptly, the neural-model $\texttt{QGen}_\theta : \mathbb{R}^d \to \mathcal{Q}$ is called the *question-generator* and the neural-model $\texttt{Gues}_\alpha : \mathbb{R}^d \to \mathcal{O}$ is called the *object-guesser*. The final neural-model $\texttt{Enc}_\beta : \mathcal{I} \times (\mathcal{Q} \times \mathcal{A})^* \to \mathbb{R}^d$ is called the *encoder* and captures pertinent features for the former models to share. Subscripts denote the parameters of each model (to be learned).

Modeling Goals There are two main objectives we consider. The first is task-oriented:

$$\min_{\alpha,\beta} \ \mathbf{E}[1\{\hat{O} \neq O\}] \tag{2}$$

which requires the predicted goal-object align with the true goal. The second objective is more elusive from a mathematical perspective: the generated dialogue $\hat{D}$ should be human-like. That is, it should be similar to the human dialogue D. As we see next, our theory is aimed at formalizing this objective.

3.2 Theoretical Framework (LEATHER)

Now, we present our proposed theory with examples from the *GuessWhat?!* game just discussed.

3.2.1 Terminology

Sets Assume a space $\mathcal{C}$, which encompasses the set of dialogue contexts, and a space $\mathcal{D}$, which encompasses the set of possible dialogues. In general, the structure of these sets and representation of elements therein are arbitrary to allow wide applicability to any dialogue system. For particular examples, consider the *Guess What?!* game: $c \in \mathcal{C}$ is an image-goal pair and $d \in \mathcal{D}$ is a list of question-answer pairs. Note, we also allow an additional, arbitrary space $\mathcal{U}$ to account for any unobserved effects on the test outputs (discussed next).

[3] By default, $m = 8$ following Shekhar et al. (2019).

[4] Notice, although the answer-player is still human, the answers may follow a distinct distribution due to dependence on the questions, so we demarcate this difference by $\tilde{\square}$.

Test Functions To evaluate generated text, we assume a group of fixed **test functions** $\{h_1 \dots h_L\}$ where for each $\ell \in [L]$ the function $h_\ell : \mathcal{D} \times \mathcal{U} \to [0, 1]$ assigns a $[0, 1]$-valued score that characterizes some high-level property of the dialogue. For example, a test function might be a binary value indicating presence of particular question-type, a continuous value indicating the proportion of clarification questions, a sentiment score, or some other user-evaluation. A test function can also be an automated metric like lexical diversity, for example.

Random Outputs As noted, the space $\mathcal{U}$ primarily allows the test h_ℓ to exhibit randomness due to unobserved effects. For example, this is the case when our test function is a human evaluation and randomness arises from the human annotator. To model this, we assume an unknown distribution $\mathbb{U}$ over $\mathcal{U}$, so that for $U \sim \mathbb{U}$ and dialogue $d \in \mathcal{D}$, the score $h_\ell(d, U)$ is a random variable. In general, we do not assume too much access to this randomness, since sampling from $\mathbb{U}$ can be costly; e.g., it can require recruiting new annotators or collecting new annotations. Note, U can also be used to encapsulate additional (observable) information needed to conduct the test h_ℓ (e.g., a reference dialogue).

Goal Distribution Next, we assume a **goal distribution** $\mathbb{G}$ over the set of contextualized dialogues; i.e., context-dialogue pairs in $\mathcal{C} \times \mathcal{D}$. Typically, $\mathbb{G}$ is the distribution of contextualized dialogues between human interlocutors. In the *GuessWhat?!* example, $\mathbb{G}$ is the distribution of the random, iterated tuple $((I, O), D)$. Recall, I is the random image and O is the random goal-object, which together form the context. $D = (Q_1, A_1 \dots Q_P, A_P)$ is the variable-length tuple of question-answer pairs produced by humans discussing the context (I, O).

Dialogue Learner and Environment We also assume some **dialogue learner** parameterized by $\theta \in \mathbb{R}^d$. The learner may only *partially* control each dialogue – e.g., the learner might only control a subset of the turns in each dialogue – and the mechanism through which this occurs is actually unimportant in the general setting; i.e., it will not be assumed in our theoretical results. Ultimately, we need only assume existence of some function $(\theta, c) \xrightarrow{\mathsf{E}} \mathbb{P}_\theta(c)$ where θ are the learned parameters, $c \in \mathcal{C}$ is the context, and $\mathbb{P}_\theta(c)$ is a distribution over dialogues $\mathcal{D}$. In the *GuessWhat?!* example discussed previously, the dialogue learner is $\texttt{QGen}_\theta$ and the function E is implicitly defined by Eq. (1). In particular, we have $\hat{D} \sim \mathbb{P}_\theta(I, O)$ where image I and object O are sampled from the goal-distribution of contextualized dialogues $((I, O), D) \sim \mathbb{G}$. We call E the **environment** of the learner and use sans serif in notation. In the *GuessWhat?!* example, the environment can change for a myriad of reasons: the answer-player could change strategies (inducing a new answer-distribution), the distribution of image I could change, or the distribution of the object O could change. All of which, can impact the function $(\theta, c) \xrightarrow{\mathsf{E}} \mathbb{P}_\theta(c)$. One implicit factor we encounter later is the dependence of the environment E on the encoder parameters β in Eq. (1). In discussion, we may explicitly write E_β to denote this dependence.

Formal Objective of Learner As discussed before, the conceptual task of the dialogue learner is to produce human-like text. To rephrase more formally: the task of the learner is to induce a contextualized dialogue distribution that is indistinguishable from the the goal distribution. Unfortunately, this objective is made difficult by the complexity of dialogue. In particular, it is unclear what features of the dialogue are important to measure: should we focus on the atomic structure of a dialogue, the overall semantics, or maybe just the fluency? Surely, the answer to this question is dependent on the application. For this reason, we suggest the general notion of a *test function*. Each test $\{h_1 \dots h_L\}$ can be hand selected prior to learning to emphasize a particular goal for the dialogue learner; e.g., as in Figure 1, h_1 can represent a user evaluation of question relevance, h_2 can capture lexical diversity, etc. Then, the quality of the contextualized dialogue distribution induced by the dialogue learner is measured by preservation of the output of the test functions. That is, the output of test functions should be similar when applied to human dialogue about the same context. We capture this idea through the **test divergence**:

$$\begin{aligned} &\mathbf{TD}_{\mathsf{E}}(\theta) = \sum\nolimits_{\ell=1}^{L} \mathbf{TD}^{\ell}_{\mathsf{E}}(\theta) \\ &\text{where} \quad \mathbf{TD}^{\ell}_{\mathsf{E}}(\theta) = \mathbf{E}[|h_\ell(D, U) - h_\ell(\hat{D}, U)|], \\ &\qquad (C, D) \sim \mathbb{G},\ \hat{D} \sim \mathbb{P}_\theta(C),\ U \sim \mathbb{U}. \end{aligned} \tag{3}$$

Notice, the test divergence is not only dependent on the parameters of the dialogue learner, but also the environment E which governs the distribution $\mathbb{P}_\theta(C)$. Recall, this function is induced by the learner's environment and its role in eliciting generated dialogue. Finally, with all terms defined, the

formal objective of the dialogue learner is typically to minimize the test divergence:

$$\min_\theta \ \mathbf{TD}_{\mathtt{E}}(\theta). \tag{4}$$

Example (BLEU/ROUGE) Useful examples of test divergence are traditional evaluation metrics, using a human reference – metrics like BLEU, ROUGE, or accuracy at next-word prediction. To see the connection, in Eq. (3), let $L = 1$, let h_1 be one of the metrics, and set $U = D$. Then, $h_1(D, U)$ computes some form of n-gram overlap between the human reference and itself, so it evaluates to 1 (full overlap). On the other hand, $h_1(\hat{D}, U)$ is the traditional notion of the metric (e.g., BLEU or ROUGE). So, the test divergence simply becomes 1 minus the average of the metric. Notice, this example shows how U can be used to encapsulate observable (random) information as well.

Example (*GuessWhat?!*) We can also consider a more complicated example in the *GuessWhat?!* game. Here, Shekhar et al. (2019) evaluate the human-likeness of dialogue with respect to the question strategies. Specifically, the authors consider a group of strategy classifiers $s_i : \mathcal{Q} \to \{0, 1\}, i \in [L]$ which each indicate presence of a particular strategy in the input question. For example, s_1 might identify if its input is a color question *"Is it blue?"* and s_2 might identify if its input is a spatial question *"Is it in the corner?"*. Then, one intuitive mathematical description of the question-strategy dissimilarity may be written

$$\mathbf{E}\left[\sum_{i=1}^{\ell}\Big|\frac{1}{P}\sum_{j=1}^{P} s_i(Q_j) - \frac{1}{m}\sum_{k=1}^{m} s_i(\hat{Q}_k)\Big|\right] \tag{5}$$

Above captures expected deviation in proportion of color/spatial questions from the human- to the generated-text. It also coincides with the definition of test divergence. To see this, note the above is Eq. (3) precisely when h_i returns the proportion of questions in a dialogue with type identified by s_i.

Example (Human Annotation) Human annotation is also an example, in which, human subjects are presented with two dialogue examples: one machine generated and one from a goal corpus with both dialogues pertaining to the same context. The human then annotates both examples with a score pertaining to the quality of the dialogue (e.g., the relevance of questions as in Figure 1). So, h_i is represented by the annotation process, using U to encapsulate any unobserved random effects. Then, the test divergence simply reports average absolute difference between annotations.

3.3 Application to a *GuessWhat?!* Algorithm

In this next part, we apply the theory just discussed to analyze a cooperative learning algorithm (`CL`) proposed by Shekhar et al. (2019). Recall Eq. (1), `CL` generates dialogue/predicted goal as below:

$$\begin{aligned}\hat{O} &= \mathtt{Gues}_\alpha(\mathtt{Enc}_\beta(I, \hat{D}))\\ \hat{Q}_{i+1} &= \mathtt{QGen}_\theta(\mathtt{Enc}_\beta(I, \hat{Q}_1, \tilde{A}_1, \ldots \hat{Q}_i, \tilde{A}_i)\end{aligned} \tag{6}$$

where $\mathtt{QGen}_\theta$ is the question-generator, $\mathtt{Gues}_\alpha$ is the object-guesser, and $\mathtt{Enc}_\beta$ is the encoder.

`CL` Algorithm Conceptually, cooperative learning encompasses a broad class of algorithms in which two or more independent model components coordinate during training to improve each other's performance. For example, this can involve a shared learning objective (Das et al., 2017). In the algorithm we consider, Shekhar et al. (2019) coordinate training of a shared encoder using two distinct learning phases. Written in the context of our theory, they are:

1. **Task-Oriented Learning**: Solve Eq. (2). Update α and β to minimize $\mathbf{E}[1\{\hat{O} \neq O\}]$.
2. **Language Learning**: Solve Eq. (4). Update θ and β to minimize $\mathbf{TD}_{\mathtt{E}_\beta}(\theta)$ where the test measures accuracy at next-word prediction.

The two phases repeat, alternating until training is finished. As is typical when training neural-networks, the parameter weights are updated using batch SGD with a differentiable surrogate loss. To do so in the **task-oriented learning phase**, $\mathtt{Gues}_\alpha$ is designed to output probability estimates for each object and the negative log-liklihood of this output distribution is minimized. In the **language learning phase**, $\mathtt{QGen}_\theta$ is designed to output probabilities for the individual utterances that compose each question. Then, the surrogate optimization is:

$$\begin{aligned}&\min_{\theta,\beta} \mathbf{E}\Big[\sum_{i+1\leq P} \mathcal{L}(\hat{Q}_{i+1}, Q_{i+1})\Big] \quad \text{where}\\ &\hat{Q}_{i+1} = \mathtt{QGen}_\theta(\mathtt{Enc}_\beta(I, Q_1, A_1 \ldots Q_i, A_i)\end{aligned} \tag{7}$$

and $\mathcal{L}$ sums the negative logliklihood of the individual utterances. Notice, a form of *teacher-forcing* is used in this objective, so that the encoder and question-generator are conditioned on *only* human dialogue during the language learning phase. This fact will become important in the next part.

Problem Importantly, the encoder parameters β are updated in *both* the *task-oriented* and *language learning* phases. So, in the language learning phase, the dialogue learner selects θ to minimize the test divergence in cooperation with a *particular* choice of the encoder parameters – let us call these β^s. Then, in the task-oriented learning phase, the learned encoder parameters may change to a new setting β^t. Importantly, by changing the parameters in Eq. (1), we induce a *new* environment $\mathsf{E}_{\beta^t} \neq \mathsf{E}_{\beta^s}$, which governs a new generation process. For brevity, we set $\mathsf{T} = \mathsf{E}_{\beta^t}$ and $\mathsf{S} = \mathsf{E}_{\beta^s}$. This change brings us to our primary issue: the shift in learning environment *does not necessarily preserve the quality of the generated dialogue*. In terms of our formal theory, we rephrase:

$$\mathbf{TD}_{\mathsf{S}}(\theta) \stackrel{?}{=} \mathbf{TD}_{\mathsf{T}}(\theta). \tag{8}$$

Without controlling the *change* in test divergence across these two environments, it is possible the two learning phases are not "cooperating" at all.

LEATHER-Inspired Solution In general, it is clear equality will not hold, but we can still ask *how different* these quantities will be. If they are very different, the quality of the dialogue generation learned in the language learning phase may degrade substantially during the task-oriented learning phase. More generally, the problem we see here is a problem of data-shift. In learning theory, the study of data-shift is often referred to as *domain adaptation*. The test divergence on the environment S – in which we learn θ – is referred to as the **source error**, while the test divergence on the environment T – in which we evaluate θ – is referred to as the **target error**. The tool we use to quantify the change between the source error and the target error is an *adaptation bound*, in which we find a statistic Δ for which the following is true:[5]

$$\mathbf{TD}_{\mathsf{T}}(\theta) \lesssim \mathbf{TD}_{\mathsf{S}}(\theta) + \Delta. \tag{9}$$

Then, we can be sure the error in the new environment has not increased much more than Δ. In this sense, we say Δ is a **predictive statistic** because it predicts the magnitude of the target error $\mathbf{TD}_{\mathsf{T}}$ from the magnitude of the source error $\mathbf{TD}_{\mathsf{S}}$. To put it more concisely, it predicts the change in error from source to target. *When Δ is small, the change should be small too or the target error should be even lower than the source error. When Δ is large, we cannot necessarily come to this conclusion.* Importantly, for Δ to be useful in practice it should not rely on too much information. In dialogue generation, it is important for Δ to avoid reliance on the *test functions*, since these can often encompass costly sampling processes like human-evaluation.

As alluded in Section 2, many adaptation bounds exist, but as it turns out, none of them are directly applicable to dialogue generation contexts. This is because, as we are aware, computation of all previous bounds relies on efficient access to the test functions $\{h_1 \dots h_L\}$ and samples $U \sim \mathbb{U}$, which is not always possible in dialogue. In particular, these functions, along with the sampling process $U \sim \mathbb{U}$, might represent a time-consuming, real-world processes like human-evaluation. For this reason, in the next section, we prove a new adaptation bound with new statistic Δ, which does not require access to the test functions.

4 Text-Generation under Data-Shift

Motivated by the *GuessWhat?!* example and algorithm CL, we continue in this section with a general study of domain adaptation for dialogue generation. We begin by proposing a new (general) adaptation bound for LEATHER. We then apply this general bound to the *GuessWhat?!* algorithm CL, motivating fruitful modifications through our analysis.

4.1 A Novel Adaptation Bound for LEATHER

The Energy Statistic and Computation

Definition 4.1. *For any independent random variables A and B, the discrete energy distance is defined $\varepsilon_{01}(A, B)$ equal to*

$$2\mathbf{E}[1\{A \neq B\}] - \mathbf{E}[1\{A \neq A'\}] - \mathbf{E}[1\{B \neq B'\}] \tag{10}$$

where A' is an i.i.d copy of A, B' is an i.i.d. copy of B, and $1\{\cdot\}$ is the indicator function; i.e., it returns 1 for true arguments and 0 otherwise.

The *discrete energy distance* is a modification of the *energy distance* sometimes called the *statistical energy*. It was first proposed by Szekely (1989) and was studied extensively by Székely and Rizzo (2013) in the case where A and B are continuous variables admitting a probability density function. In general, and especially in dialogue, this is not the case. Aptly, our newly suggested form of the energy distance is more widely applicable to any

[5] The inequality is approximate because there are often other statistics in the bound, but through reasonable assumptions, one statistic Δ is identified as the key quantity of interest. These assumptions should be carefully made to avoid undesirable results (Ben-David et al., 2010b; Zhao et al., 2019).

variables A and B for which equality is defined. While general, this distance can be insensitive, especially when A and B take on many values. To remedy this, we introduce the following.

Definition 4.2. *Let $\mathcal{D}$ be any set. A coarsening function is a map $c : \mathcal{D} \to \mathcal{D}$ such that $c(\mathcal{D}) = \{c(d) \mid d \in \mathcal{D}\}$ is finite, and further, $|c(\mathcal{D})| < |\mathcal{D}|$.*

Since $\mathcal{D}$ is likely an immensely large set, this can make the signal $1\{a \neq b\}$ for $a, b \in \mathcal{D}$ overwhelming compared to the signal $1\{a = b\}$, and therefore, weaken the sensitivity of the discrete energy distance, overall. Coarsening functions allow us to alleviate this problem by effectively "shrinking" the set $\mathcal{D}$ to a smaller set. To do this, the role of the coarsening function is to exploit additional context to arrive at an appropriate *clustering* of the dialogues, which assigns conceptually "near" dialogues to the same cluster. So, the choice of $c(d)$ should be a "good" representation of d, in the sense that too much valuable information is not lost. As a general shorthand, for a coarsening function c and variables A, B, we write

$$\varepsilon_c(A, B) = \varepsilon_{01}(c(A), c(B)). \tag{11}$$

In this paper, we implement c using the results of a k-means clustering with details in Appendix A.

Adaptation Bound With these defined, we give the novel bound. Proof of a more general version of this bound – applicable beyond dialogue contexts (e.g., classification) – is provided in Appendix B Thm. B.1. Notably, our proof requires some technical results on the relationship between discrete energy and the characteristic functions of discrete probability distributions. These may also be of independent interest, outside the scope of this paper.

Theorem 4.1. *For any $\theta \in \mathbb{R}^d$, any coarsening function $c : \mathcal{D} \to \mathcal{D}$, and all $\ell \in [L]$*

$$\mathbf{TD}^{\ell}_{\mathsf{T}}(\theta) \leq \gamma + \varphi + \mathbf{TD}^{\ell}_{\mathsf{S}}(\theta) + \sqrt{\varepsilon_c(\tilde{D}_1, \tilde{D}_2) \times \delta} \tag{12}$$

where $\tilde{D}_1 \sim \mathbb{P}_\theta(C) = \mathsf{T}(\theta, C)$, $\tilde{D}_2 \sim \mathbb{Q}_\theta(C) = \mathsf{S}(\theta, C)$, $(C, D) \sim \mathbb{G}$, $U \sim \mathbb{U}$,[6]

$$\begin{aligned}
\gamma &= \sum\nolimits_{i \in \{1,2\}} \mathbf{E}[|h_\ell(c(\tilde{D}_i), U) - h_\ell(\tilde{D}_i, U)|] \\
g &\in \operatorname*{arg\,min}_{f \in [0,1]^{\mathcal{D} \times \mathcal{U}}} \sum\nolimits_i \mathbf{E}[|f(c(\tilde{D}_i), U) - h_\ell(D, U)|] \\
&\quad \textit{where} \quad [0,1]^{\mathcal{D} \times \mathcal{U}} = \{f \mid f : \mathcal{D} \times \mathcal{U} \to [0,1]\}. \\
\varphi &= \sum\nolimits_{i \in \{1,2\}} \mathbf{E}[|g(c(\tilde{D}_i), U) - h_\ell(D, U)|] \\
\delta &= \mathbf{E}\Big[\sum\nolimits_{x \in c(\mathcal{D})} |g(x, U) - h_\ell(x, U)|\Big].
\end{aligned} \tag{13}$$

[6]For simplicity, let $\tilde{D}_1, \tilde{D}_2, U$ be pairwise-independent.

Unobserved Terms in Dialogue As noted, an important benefit of our theory is that we need not assume computationally efficient access to the test functions $\{h_1 \dots h_L\}$ or samples $U \sim \mathbb{U}$. Yet, the reader likely notices a number of terms in Eq. (12) dependent on both of these. Similar to the traditional case, we argue that our theory is still predictive because it is often appropriate to assume these unobserved terms are small, or otherwise irrelevant. We address each of them in the following:

1. The term γ captures average change in test output as a function of the coarsening function c. Whenever $c(\tilde{D}_i)$ is a good representative of $\tilde{D}_i$ (i.e., it maintains information to which h_ℓ is sensitive) γ should be small.
2. The next term φ is the smallest sum of expected differences that *any* function of the coarsened dialogues $c(\tilde{D}_i)$ and the arbitrary randomness U can achieve in mimicking the true test scores $h_\ell(D, U)$. Since the set of all functions from $\mathcal{D} \times \mathcal{U}$ to $[0, 1]$ should be very expressive, this can be seen as another requirement on our coarsened dialogues $c(\tilde{D}_i)$. For example, when $c(\tilde{D}_i) = \tilde{D}_i \approx D$ this term can be close to zero. When instead $|c(\mathcal{D})|$ is much smaller than $|\mathcal{D}|$ (e.g., a singleton set), we expect φ to grow.
3. The last term δ can actually be large. Fortunately, since δ is multiplied by the energy distance, this issue is mitigated when the statistical energy is small enough. Ultimately, the energy is paramount in controlling the impact of this term on the bound's overall magnitude.

A Predictive Theory Granted the background above, our discussion reduces the predictive aspect of the bound to a single key quantity: the discrete energy distance $\varepsilon_c(\tilde{D}_1, \tilde{D}_2)$. In particular, besides the test divergence $\mathbf{TD}_{\mathsf{S}}$, all other terms can be assumed reasonably small by proper choice of the coarsening function, or otherwise controlled by the statistical energy through multiplication. Note, the first issue is discussed in Appendix A. Ultimately, the main takeaway is that statistical energy plays the role of Δ as discussed in Section 3.3.

4.2 A New Cooperative Learning Algorithm

With all theoretical tools in play, we return to the algorithm `CL` and the problem raised in Section 3.3.

`LEATHER`-Motivated Modification Recall, we are interested in quantifying and controlling the change in error from source $\mathbf{TD}_{\mathsf{S}}(\theta)$ to target $\mathbf{TD}_{\mathsf{T}}(\theta)$ across the training phases. Based on our

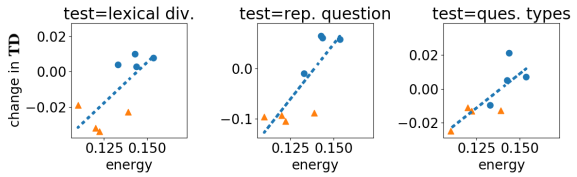

Figure 2: Energy between training phases. Energy is predictive of change in test divergence as desired. Dotted line is line of best fit. Blue circles (CL) indicate use of *only* generated dialogue in task-oriented learning phase. Orange triangles (LEATHER) indicate regularization with human data.

theory, we know we should decrease the statistical energy between dialogues to reduce this change. That is, we should reduce the distance between the generated dialogue distributions across learning phases. We hypothesize this may be done by incorporating human dialogue in the task-oriented learning phase. The encoder in CL sees *no* human dialogue when forming the prediction $\hat{O}$ that is compared to O during task-oriented learning – as seen in Eq. (1), only the generated dialogue $\hat{D}$ is used. In contrast, the encoder sees *only* the human dialogue D in the alternate language learning phase – i.e., as seen in the surrogate objective in Eq. (7). We hypothesize this stark contrast produces large shifts in the parameters $\beta^s \rightarrow \beta^t$ between phases. Instead, we propose to *regularize* the task-oriented learning phase with human dialogue as below:

$$\begin{aligned} &\min_{\alpha,\beta} \mathbf{E}[1[\hat{O} \neq O]] + \mathbf{E}[1[\hat{O}' \neq O]] \quad \text{where} \\ &\hat{O}' = \texttt{Gues}_\alpha(\texttt{Enc}_\beta(I, D)), \quad ((I, O), D) \sim \mathbb{G} \end{aligned} \tag{14}$$

and $\hat{O}$ is still as described in Eq. (1). Intuitively, this should constrain parameter shift from $\beta^s \rightarrow \beta^t$, thereby constraining the change in outputs of the encoder, and ultimately constraining the change in outputs of the question-generator, which is conditioned on the encoder outputs. As the generated dialogue distributions from distinct learning phases will be more similar by this constraint, we hypothesize the penultimate effect will be decreased statistical energy (i.e., since energy measures distance of distributions). Based on our theory, reduced energy provides resolution to our problem: test divergence should be preserved from source to target.

5 Experiments

5.1 Cooperative Learning via LEATHER

Setup In general, we use experimental settings of Shekhar et al. (2019) (e.g., hyperparameters, validation, etc.) with full details available in the code. CL denotes the original algorithm proposed by Shekhar et al. (2019) (Section 3.3). LEATHER denotes our LEATHER-inspired modification (Section 4.2).

Automated Metrics We report average accuracy **acc** of the guesser module in identifying the true goal-object across three random seeds as well as average lexical diversity (**lexdiv**; type/token ratio over all dialogues), average question diversity (**qdiv**; % unique questions over all dialogues), and average percent of dialogues with verbatim repeated questions (**repq**). **acc** quantifies task-success, while subsequent metrics are designed to quantify human-likeness of the generated dialogue. These metrics were all previously computed by Shekhar et al. (2019) with details in their code.

Human Evaluation We asked two annotators to help us further evaluate the results. Throughout the process, human subject guidelines from the authors' institution were followed and the task was approved by our institution human subject board. The annotators examined contextualized human dialogues and generated dialogues from a CL model and LEATHER model. All dialogues used the same image/goal context and annotators observed all dialogues for a specific context in random order without knowing how each dialogue was created. Across 50+ dialogues, average percentage of irrelevant questions per dialogue (**irrq**) was determined.[7] Average percentage of specific questions (**spcq**) was also determined.[8] We report **TD**, which gives the average *difference* in percentages from the corresponding human dialogue. Sans scaling, these **TD** metrics are examples of the test divergence in Eq. (3) using a human-evaluation test function. Qualitative analysis of errors was also conducted based on annotator remarks (provided later in this section).

Impact of LEATHER In Table 1, we compare the cooperative learning algorithms CL and LEATHER. The former uses only the generated dialogue during task-oriented learning, while the latter incorporates human data to regularize the change in parameters underlying the environmental shift. As predicted by our theory, regularization is very beneficial, improv-

[7] An *irrelevant* question ignores the image or current dialogue context. For example, in Figure 1, CL asks about the man's "face" (Q5) after learning the goal-object is a car, which ignores dialogue-context. CL also hallucinates an object "cut off" on the right side (Q4), which ignores image context.

[8] A *specific* question contains two or more modifiers of one or more nouns. For example, LEATHER modifies "car" with "behind" and "man" with "the white shirt" in Figure 1 Q7.

	acc ↑	lexdiv ↑	qdiv ↑	repq ↓	irrq(TD) ↓	spcq(TD) ↓	energy ↓
CL	57.1 (55.9)	9.98 (10.7)	13.5 (14.3)	55.9 (58.2)	30.5	23.3	0.143
LEATHER	58.4 (56.9)	11.4 (12.7)	13.1 (16.0)	53.6 (47.5)	26.2	19.5	0.123
RL	56.3	7.3	1.04	96.5	-	-	-

Table 1: Comparison of CL and our theory-motivated modification LEATHER. Best epoch based on validation **acc** is reported with last epoch in parentheses. Up/down arrows indicate objective. Metrics are on 100 point scale, excluding **energy**. The first 4 metrics are automated, the next 2 are from human evaluation, and the last is our proposed statistic. LEATHER improves accuracy and human-likeness of dialogue. Further, our proposed statistic **energy** is predictive of human-likeness.

ing task-success and human-likeness. For example, LEATHER decreases % of irrelevant questions by 4.8% compared to CL, which is more similar to human dialogue according to the test divergence (**TD**). Interestingly, LEATHER also decreased % of specific questions by 1.7%. Based on the **TD**, this is *also* more similar to human dialogue, indicating humans ask fewer specific questions too. The design of the **TD** allows us to capture these non-intuitive results. Notably, regularization inspired by LEATHER *allows us to train longer* without degrading task-success or suffering from mode collapse (i.e., repeated questions). Automated human-likeness metrics for the last epoch (in parentheses) show substantial improvements over CL in this case.

Cooperative vs. Reinforcement Learning In Table 1, we compare the two cooperative learning algorithms CL and LEATHER to the reinforcement learning algorithm (RL). We use the results reported by Shekhar et al. (2019) for RL, since we share an experimental setup. Compared to RL, both cooperative learning approaches improve task success and human-likeness. As noted in Section 2, the theoretical framework for RL (i.e., POMDPs) is not equipped to study interaction of the distinct learning phases within this algorithm (i.e., with respect to data-shift). Better theoretical understanding could explain poor performance and offer improvement as demonstrated with LEATHER, which improves human-likeness of CL.

Qualitative Analysis In dialogue generated by CL, questions with poor relevance ignored the image context (e.g., model hallucination). In dialogue generated by the LEATHER model, irrelevant questions ignored current dialogue context (e.g., a question which should already be inferred from existing answers). We hypothesize this may be due to poor faith in the automated answer-player used for training, which also has problems with model hallucination (e.g., Figure 1). Both models had issues with repeated questions. In human dialogue, issues were grammatical with few irrelevant questions.

5.2 LEATHER is Empirically Predictive

Here, we show statistical energy predicts test divergence, empirically. Computation of energy can be automated, so predictive ability is useful for model-selection when human evaluation is not available. We consider test divergence (**TD**) with 4 groups of tests: (**A**) the 9 fine-grained strategy classifiers of Shekhar et al. (2019) used as in Eq. (5), (**B**) lexical diversity computed as type/token ratio per dialogue, (**C**) question repetition computed as a binary indicator for each dialogue, and (**D**) the discussed human-evaluations of question relevance/specificity. Figure 2 plots change in **TD** for (**A-C**) as a function of energy. Specifically, change in **TD** is the difference $\mathbf{TD}_{\mathsf{T}}(\theta) - \mathbf{TD}_{\mathsf{S}}(\theta)$ where S and T are defined by the transition from language learning to task-oriented learning discussed in Section 3. We plot this change at the transitions after epochs 65, 75, 85, and 95 (out of 100 total). Notably, *energy is predictive and, specifically, is linearly related to change in test divergence.* For (**D**), in Table 1, we show average energy across all transitions compared to test divergence. Energy is also predictive for these human-evaluation tests.

6 Conclusion

This work presents LEATHER, a theoretically motivated framework for learning to generate human-like dialogue. The energy statistic, which is derived from this theory, is used to analyze *and improve* an algorithm for task-oriented dialogue generation. Further, energy is empirically predictive of improvements in dialogue quality, measured by both automated and human evaluation. Future work may involve more experiments to test the utility of LEATHER in other dialogue settings. Theoretically, we hope to study sample-complexity in LEATHER, which is a hallmark of common PAC theories.

Acknowledgments

We thank the anonymous reviewers for helpful feedback and Jennifer C. Gates, CCC-SLP, for input on qualitative evaluations of dialogue in experiments.

References

Katherine Atwell, Anthony Sicilia, Seong Jae Hwang, and Malihe Alikhani. 2022. The change that matters in discourse parsing: Estimating the impact of domain shift on parser error. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 824–845.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010a. A theory of learning from different domains. *Machine learning*, 79(1):151–175.

Shai Ben-David, Tyler Lu, Teresa Luu, and Dávid Pál. 2010b. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136. JMLR Workshop and Conference Proceedings.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Elia Bruni and Raquel Fernandez. 2017. Adversarial evaluation for open-domain dialogue generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 284–288.

R Cuppens. 1975. Decomposition of multivariate distributions.

Abhishek Das, Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. 2017. Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2951–2960.

Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5503–5512.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. 2020. Pac-bayes and domain adaptation. *Neurocomputing*, 379:379–397.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Mert Inan, Piyush Sharma, Baber Khalid, Radu Soricut, Matthew Stone, and Malihe Alikhani. 2021. Cosmic: A coherence-aware generation metric for image descriptions. *arXiv preprint arXiv:2109.05281*.

Fredrik D Johansson, David Sontag, and Rajesh Ranganath. 2019. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 527–536. PMLR.

Sham Machandranath Kakade. 2003. *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom).

Seiichi Kuroki, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato, and Masashi Sugiyama. 2019. Unsupervised domain adaptation based on source-guided discrepancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4122–4129.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. 2018. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

María Pérez-Ortiz, Omar Rivasplata, John Shawe-Taylor, and Csaba Szepesvári. 2021. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22.

Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. 2019. Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems*, 32.

Ievgen Redko, Amaury Habrard, and Marc Sebban. 2017. Theoretical analysis of domain adaptation with optimal transport. In *ECML PKDD*, pages 737–753. Springer.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. 2020. A survey on domain adaptation theory. *ArXiv*, abs/2004.11829.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Ravi Shekhar, Aashish Venkatesh, Tim Baumgärtner, Elia Bruni, Barbara Plank, Raffaella Bernardi, and Raquel Fernández. 2019. Beyond task success: A

closer look at jointly learning to see, ask, and GuessWhat. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2578–2587, Minneapolis, Minnesota. Association for Computational Linguistics.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. 2018. Wasserstein distance guided representation learning for domain adaptation. In *AAAI*.

Anthony Sicilia, Katherine Atwell, Malihe Alikhani, and Seong Jae Hwang. 2022a. Pac-bayesian domain adaptation bounds for multiclass learners. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

Anthony Sicilia, Tristan Maidment, Pat Healy, and Malihe Alikhani. 2022b. Modeling non-cooperative dialogue: Theoretical and empirical insights. *Transactions of the Association for Computational Linguistics*, 10:1084–1102.

Florian Strub, Harm De Vries, Jeremie Mary, Bilal Piot, Aaron Courvile, and Olivier Pietquin. 2017. End-to-end optimization of goal-driven and visually grounded dialogue systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 2765–2771.

Gabor J Szekely. 1989. Potential and kinetic energy in statistics. *Lecture Notes, Budapest Institute*.

Gábor J Székely and Maria L Rizzo. 2013. Energy statistics: A class of statistics based on distances. *Journal of statistical planning and inference*, 143(8):1249–1272.

Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J Gordon. 2020. Domain adaptation with conditional distribution matching and generalized label shift. *Advances in Neural Information Processing Systems*, 33:19276–19289.

Leslie G Valiant. 1984. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019a. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019b. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR.

Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. 2019. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pages 7523–7532. PMLR.

A Novel Adaptation Bound and Computation of Energy Statistic

In this section, we give our novel adaptation bound and details for the accompanying energy statistic. There is some redundancy between this section and Section 4, but in general, this section is more detailed. Recall, *source* error is denoted $\mathbf{TD}_{\mathsf{S}}$ and is observed on the environment $\mathbb{Q}_\theta(c) = \mathsf{S}(\theta, c)$. The *target* error is denoted $\mathbf{TD}_{\mathsf{T}}$ and is observed on the environment $\mathbb{P}_\theta(c) = \mathsf{T}(\theta, c)$. For the algorithm CL discussed in the main text, the target is induced by the task-oriented learning phase and the source is induced by the language learning phase.

A.1 The Problem with Traditional Bounds

Predictive Adaptation Theories An important quality of traditional domain adaptation bounds, proposed for classification and regression problems, is that they offer a *predictive theory*. Namely, without observing the target error $\mathbf{TD}_{\mathsf{T}}$, we can infer this quantity from Δ and the source error $\mathbf{TD}_{\mathsf{S}}$. The utility of this is two-fold: first, it allows us to design algorithms that prepare a learner for data-shift by controlling Δ; second, it allows a practitioner to select an appropriate model to deploy in the presence of data-shift by comparing the different values of Δ for each model. In general, these use-cases would not be possible without Δ because the target error $\mathbf{TD}_{\mathsf{T}}$ *is not observable until it is too late.* In contrast, the quantity Δ *should* be observable. While this is not always true of Δ, authors typically reduce the main effect of Δ to one key statistic, which *is* observable. For example, Atwell et al. (2022) reduce Δ to one key statistic called the h-discrepancy by suggesting the other components making up Δ are small. This is why we use an "approximate" inequality in the main text, since other (small) terms may contribute to the bound.

Traditional Theories Are Not Predictive Traditional theories of adaptation are *not* predictive for dialogue generation. Namely, computation of Δ and its key components generally relies on computationally efficient access to the tests $\{h_1 \dots h_L\}$ and requires sampling from the unknown distribution $U \sim \mathbb{U}$. While we can always *observe* the outputs of $\{h_1 \dots h_L\}$ with randomness $U \sim \mathbb{U}$ through the source error $\mathbf{TD}_{\mathsf{S}}(\theta)$, it is *not* always the case that we have computational efficiently access to these tests or the randomness. For example, as noted in Section 3.2.1, the group of tests $\{h_1 \dots h_L\}$ along with samples U from the unknown distribution $\mathbb{U}$ may represent complex real-world processes such as human-evaluation. Even for simpler evaluation metrics based on text-classifiers (e.g., like $\{s_1 \dots s_L\}$ in Eq. (5)) algorithms for computing Δ turn out to be non-trivial, and must be handled on a case-by-case basis. Thus, in generation contexts, we typically have no way of computing Δ algorithmically, and when we do, it can be difficult to implement. If we require an easily implemented, predictive theory, then the classical theory is ruled out. As a solution, we propose a novel adaptation bound.

A.2 A Novel Adaptation Bound

First, we define some terms.

The Energy Statistic and Computation

Definition A.1. *For any independent random variables A and B, the discrete energy distance is defined:*

$$\varepsilon_{01}(A, B) = 2\mathbf{E}[1\{A \neq B\}] - \mathbf{E}[1\{A \neq A'\}] - \mathbf{E}[1\{B \neq B'\}] \tag{15}$$

where A' is an i.i.d copy of A, B' is an i.i.d. copy of B, and $1\{\cdot\}$ is the indicator function; i.e., it returns 1 for true arguments and 0 otherwise.

The *discrete energy distance* is a modification of the *energy distance* sometimes called the *statistical energy*. It was first proposed by Szekely (1989) and was studied extensively by Székely and Rizzo (2013) in the case where A and B are continuous variables admitting a probability density function. In general, and especially in dialogue, this is not the case. Aptly, we suggest the above form of the energy distance, which is widely applicable to any variables A and B for which equality is defined. While general, this energy distance can be strict and insensitive, especially when A and B take on many possible values. To remedy this, we propose the following addendum.

Definition A.2. *Let $\mathcal{D}$ be any set. A coarsening function is a map $c : \mathcal{D} \to \mathcal{D}$ such that $c(\mathcal{D}) = \{c(d) \mid d \in \mathcal{D}\}$ is finite, and further, $|c(\mathcal{D})| < |\mathcal{D}|$.*

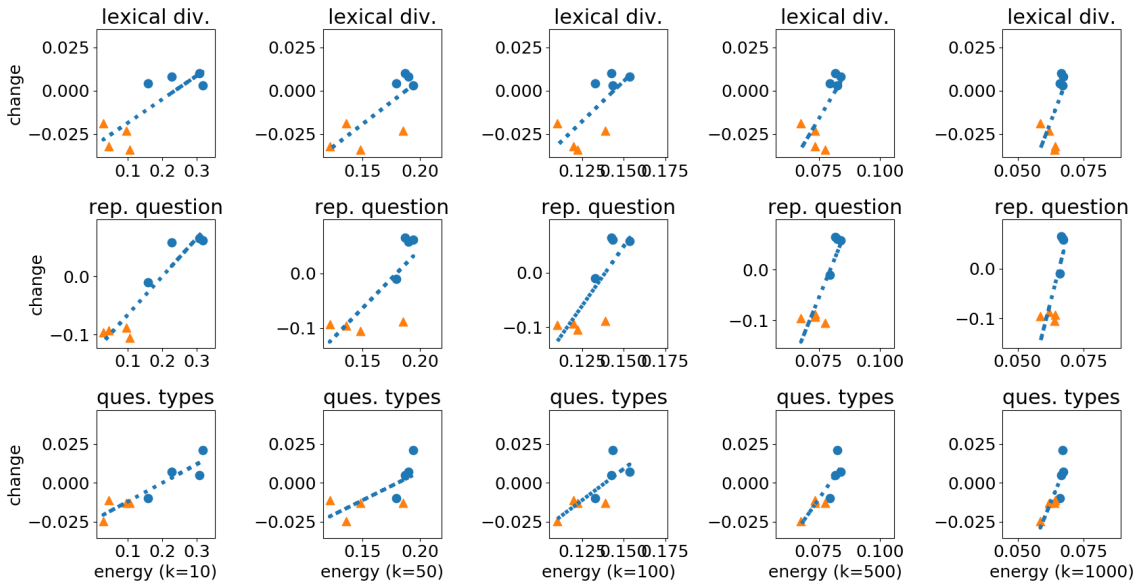

Figure 3: Comparison of energy statistics and automated test functions as in Section 5. Here, we vary the parameter k in the k-means clustering used to determine the *coarsening function* when computing energy. Trends reported in the main text are robust to variation in k.

Since $\mathcal{D}$ is likely an immensely large set, this can make the signal $1\{a \neq b\}$ for $a, b \in \mathcal{D}$ overwhelming compared to the signal $1\{a = b\}$, and therefore, weaken the sensitivity of the discrete energy distance, overall. Coarsening functions allow us to alleviate this problem by effectively "shrinking" the set $\mathcal{D}$ to a smaller set. To do this, the role of the coarsening function is to exploit additional context to arrive at an appropriate *clustering* of the dialogues, which assigns conceptually "near" dialogues to the same cluster. So, the choice of $c(d)$ should be a "good" representation of d, in the sense that too much valuable information is not lost. As a general shorthand, for a coarsening function c and variables A, B, we write

$$\varepsilon_c(A, B) = \varepsilon_{01}(c(A), c(B)). \tag{16}$$

Example One example of a coarsening function for dialogues is k-means clustering. In fact, this is the coarsening function we use to compute energy in Section 5, selecting $k = 100$. Real-valued vector representations of dialogues (e.g., from model latent space) can capture semantic information about the dialogue (Bowman et al., 2015), so we use latent space representations (i.e., the output of the encoder) to represent each dialogue and conduct a k-means clustering on these representations. For a dialogue d the output $c(d)$ is then defined by the cluster of d; i.e., we select an arbitrary dialogue to represent the whole of each cluster and assign this dialogue as the output $c(d)$. In practical implementations, it is typically easier to just compute the energy distance on the cluster labels themselves; this statistic is always equivalent to the energy on the coarsened dialogues, since the map between cluster representatives and cluster labels is bijective. Later, within Lemma B.3, we prove this equivalence for any bijective map.

Of course, regardless of implementation, this clustering is dependent on the choice of k. Figure 3 shows that the results in Section 5 are robust to different choices of k. In all cases, there is a linear relationship between the energy and the change in the test divergence.

Adaptation Bound With these defined, we give the novel bound. Proof of a more general version of this bound – applicable beyond dialogue contexts – is provided in Appendix B Thm. B.1. In particular, the general version is "backwards compatible" in the sense that it also applies to traditional learning theoretic settings like classification and regression. Arguably, in these settings, it also remains more computationally efficient than existing theories. Notably, our proof requires some technical results on the relationship between discrete energy and the characteristic functions of discrete probability distributions. These may also be of independent interest, outside the scope of this paper.

Theorem A.1. *For any $\theta \in \mathbb{R}^d$, any coarsening function $c : \mathcal{D} \to \mathcal{D}$, and all $\ell \in [L]$*

$$\mathbf{TD}^{\ell}_{\mathsf{T}}(\theta) \leq \gamma + \varphi + \mathbf{TD}^{\ell}_{\mathsf{S}}(\theta) + \sqrt{\varepsilon_c(\tilde{D}_1, \tilde{D}_2) \times \delta} \tag{17}$$

where $\tilde{D}_1 \sim \mathbb{P}_\theta(C) = \mathsf{T}(\theta, C),\ \tilde{D}_2 \sim \mathbb{Q}_\theta(C) = \mathsf{S}(\theta, C),\ (C, D) \sim \mathbb{G},\ U \sim \mathbb{U}$,[9]

$$\begin{aligned}
&\gamma = \mathbf{E}[|h_\ell(c(\tilde{D}_1), U) - h_\ell(\tilde{D}_1, U)|] + \mathbf{E}[|h_\ell(c(\tilde{D}_2), U) - h_\ell(\tilde{D}_2, U)|] \\
&g \in \operatorname*{arg\,min}_{f \in [0,1]^{\mathcal{D} \times \mathcal{U}}} \sum\nolimits_i \mathbf{E}[|f(c(\tilde{D}_i), U) - h_\ell(D, U)|] \quad \textit{where} \quad [0,1]^{\mathcal{X} \times \mathcal{U}} = \{f \mid f : \mathcal{X} \times \mathcal{U} \to [0,1]\}. \\
&\varphi = \mathbf{E}[|g(c(\tilde{D}_1), U) - h_\ell(D, U)|] + \mathbf{E}[|g(c(\tilde{D}_2), U) - h_\ell(D, U)|] \\
&\delta = \mathbf{E}\Big[\sum\nolimits_{x \in c(\mathcal{D})} |g(x, U) - h_\ell(x, U)|\Big].
\end{aligned} \tag{18}$$

Unobserved Terms in Dialogue As noted, an important benefit of our theory is that we need not assume computationally efficient access to the test functions $\{h_1 \dots h_L\}$ or samples $U \sim \mathbb{U}$. Yet, the reader likely notices a number of terms in Eq. (17) dependent on both of these. Similar to the traditional case, we argue that our theory is still predictive because it is typically appropriate to assume these unobserved terms are small, or otherwise irrelevant. We address each of them in the following:

1. The term γ captures average change in test output as a function of the coarsening function c. Whenever $c(\tilde{D}_i)$ is a good representative of $\tilde{D}_i$ (i.e., it maintains information to which h_ℓ is sensitive) γ should be small. Since we choose the coarsening function, the former premise is not a strong requirement. In practice, if choice of c is unclear, we recommend studying many choices as in Figure 3.
2. The next term φ is the smallest sum of expected differences that *any* function of the coarsened dialogues $c(\tilde{D}_i)$ and the arbitrary randomness U can achieve in mimicking the true test scores $h_\ell(D, U)$. In general, the set of all functions from $\mathcal{D} \times \mathcal{U}$ to $[0, 1]$ should be very expressive; e.g., it contains h_ℓ itself and any other function which might mimic $h_\ell(D, U)$ better when applied to $c(\tilde{D}_i)$ and U. So, it is not unreasonable to expect some good minimizer to exist, and therefore, φ to be small. Using this logic, one additional constraint is that $c(\tilde{D}_i)$ has appropriate variance. For instance, if $c(\tilde{D}_i)$ is constant and D is not, φ can easily be large. Instead, when $c(\tilde{D}_i)$ does have variance, the expressiveness of the function class $[0, 1]^{\mathcal{D} \times \mathcal{U}}$ can be well exploited. For reasonable dialogue learners and a well-chosen c, the variance of $c(\tilde{D}_i)$ is a non-issue.
3. The last term δ may actually be large, but we argue this is also a non-issue for interpretation purposes. In general, because δ is an *unnormalized* sum, its magnitude grows with the size of $c(\mathcal{D})$, even if the individual summands may be small. Fortunately, since δ is multiplied by the energy distance, this issue is mitigated when the statistical energy is small enough. Ultimately, the energy is paramount in controlling the impact of this term on the bound's overall magnitude.

A Predictive Theory Granted the background above, our discussion reduces the predictive aspect of the bound to a single key quantity: the discrete energy distance $\varepsilon_c(\tilde{D}_1, \tilde{D}_2)$. In particular, besides the test divergence $\mathbf{TD}_{\mathsf{S}}$ (known prior to the environmental change), all other terms can be assumed reasonably small, or otherwise controlled by the statistical energy through multiplication. Therefore, *if the statistical energy between environments is small, it can be reasonable to assume the dialogue quality has been maintained or improved. Otherwise, it is possible the quality of the generated dialogue has substantially degraded.* In this way, the statistical energy is an easily observable quantity that assists us in determining if the source error $\mathbf{TD}_{\mathsf{S}}$ known before the environmental change is a good representative of the unknown target error $\mathbf{TD}_{\mathsf{T}}$, which is observed after the environmental change.

Use Cases In general, controlling the statistical energy between dialogues ensures we preserve dialogue quality when the evaluation metrics we care about are not available. As demonstrated in the main text, this makes it useful in algorithm design; i.e., to inform decisions in model training. Energy can also be useful for model selection. Namely, the generation model whose dialogues have the smallest energy compared to goal dialogue should produce the highest quality dialogue. To see this, simply set $\tilde{D}_2 = D$ in the bound. Similar logical reduction shows the energy is the dominating term in this case as well.

[9] For simplicity, let $\tilde{D}_1, \tilde{D}_2, U$ be pairwise-independent. When independence does not hold, similar results can be derived under assumption of context-conditional independence.

B Proofs

In this section we prove the claimed theoretical results. So that the results may be more broadly applicable, we prove them in a more general context and then specify to the context of dialogue generation (in the main text and Appendix A).

B.1 An Adaptation Bound Based on a Discrete Energy Statistic

In this section, we propose an adaptation bound based on the energy statistic. As we are aware, ours are the first theoretical results relating the statistical energy between distributions to the change in function outputs across said distributions. Given the use of the discrete energy distance (Def. A.1) and the accompanying coarsening function (Def. A.2), we appropriately choose to prove our theoretical results for discrete random variables (i.e., those which take on only a countable number of values and exhibit a probability mass function). The effect of this choice is that we also contribute a number of new theoretical results relating the probability mass function of a real-valued, discrete random variable to its characteristic function (i.e., in similar style to the Parseval-Plancherel Theorem). Furthermore, we expand on the relationship between the statistical energy of distributions and their characteristic functions. While this has been well studied in the continuous setting (Székely and Rizzo, 2013) where the distributions of random variables admit probability densities (i.e., absolutely continuous with respect to the Lesbesgue measure), it has not been studied in the case of discrete random variables. We start our results using only *real-valued* discrete variables, but prove our main results for *all* discrete random variables using Lemma B.3

B.1.1 Setup

Suppose A and B are discrete random variables taking on values in $\mathbb{R}^d$ for some d. Respectively, the distribution of A is α and the distribution of B is β. The space $\Omega \subset \mathbb{R}^d$ is the countable subset of $\mathbb{R}^d$ for which α or β assigns non-zero probability; i.e., $\Omega = \text{supp}(\alpha) \cup \text{supp}(\beta)$. Then, the expectation of any function $f : \mathbb{R}^d \to \mathbb{R}$ of A is defined:

$$\mathbf{E}[f(A)] = \int_{\mathbb{R}^d} f \mathrm{d}\alpha = \sum_{a \in \Omega} f(a) p_\alpha(a) \tag{19}$$

where p_α is the probability mass function for A (i.e., α). Expectations of functions of B are similarly defined.

The *characteristic function* of A is defined as the complex-conjugate of the Fourier-Stieltjes transform of the probability mass function p_α. More explicitly, it is the function $\hat{p}_\alpha : \mathbb{R}^d \to \mathbb{R}$ defined

$$\hat{p}_\alpha(\tau) = \mathbf{E}[\exp\{i\tau^{\mathrm{T}} A\}] = \sum_{a \in \Omega} p_\alpha(a) \exp\{i\tau^{\mathrm{T}} a\} \tag{20}$$

where i is the imaginary unit (i.e., $i^2 = -1$) and $\tau^{\mathrm{T}} a$ is the (inner) product between column vectors τ and a. Note, the characteristic function always exists and is finite for each τ.

B.1.2 Parseval-Plancherel Theorem (Reprise)

One notable use for the *characteristic function* is the following *inversion formula*. In the discrete context we consider, Cuppens (1975) proves the following

$$p_\alpha(a) = \lim_{\tau_1 \to \infty} \lim_{\tau_2 \to \infty} \cdots \lim_{\tau_d \to \infty} \left(\prod_{i=1}^{d} 1/(2\tau_i) \right) \int_{B(\tau)} \hat{p}_\alpha(t) \exp\{-it^{\mathrm{T}} a\} \lambda(\mathrm{d}t) \tag{21}$$

where $\tau = (\tau_1, \tau_2, \ldots, \tau_d)^{\mathrm{T}}$, $B(\tau) = \{x \in \mathbb{R}^d \mid -\tau_i \leq x_i \leq \tau_i\}$, and λ is the Lebesgue measure. This inversion formula highlights the connection between the characteristic function and the general Fourier transform as alluded to just before Eq. (20), since Fourier transforms are well known for their own inversion formulas. Another commonly used result in Fourier Analysis (related to inversion) is the Parseval-Plancherel Theorem. We prove a variation on this result below. As we are aware, it is the first which uses the transform given in Eq. (20) (i.e., specific to discrete, real-valued random variables).

Lemma B.1. *For any discrete random variables A and B as described, taking values in $\mathbb{R}^d$,*

$$\sum_{x\in\Omega}|p_\alpha(x)-p_\beta(x)|^2=\lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\dots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\int_{B(\tau)}|\hat{p}_\alpha(t)-\hat{p}_\beta(t)|^2\lambda(\mathrm{d}t). \tag{22}$$

Proof. For any function $f:\mathbb{R}^d\to\mathbb{R}^+$ such that $\sum_{x\in\Omega}f(x)<\infty$ for all $t\in\mathbb{R}^d$, we prove the following more general result

$$\sum_{x\in\Omega}f^2(x)=\lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\dots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\int_{B(\tau)}\hat{f}(x)\hat{f}^*(x)\lambda(\mathrm{d}t) \tag{23}$$

where as before a "hat" denotes the Fourier-Stieltjes transform given in Eq. (20) and the new notation $\hat{f}^*$ denotes the complex-conjugate of $\hat{f}$. Observe, this proves the desired results because setting $f(x)=p_\alpha(x)-q_\alpha(x)$ we have

$$f^2(x)=(p_\alpha(x)-q_\alpha(x))^2=|p_\alpha(x)-q_\alpha(x)|^2 \tag{24}$$

and

$$\begin{aligned}\hat{f}(x)\hat{f}^*(x)&=(\widehat{p_\alpha(x)-p_\alpha}(x))(\widehat{p_\alpha(x)-p_\alpha}(x))^*\\&=(\hat{p}_\alpha(x)-\hat{p}_\alpha(x))(\hat{p}_\alpha(x)-\hat{p}_\alpha(x))^*=|\hat{p}_\alpha(x)-\hat{p}_\alpha(x)|^2.\end{aligned} \tag{25}$$

Proceeding with the proof of Eq. (23) we have

$$\begin{aligned}&\lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\dots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\int_{B(\tau)}\hat{f}(x)\hat{f}^*(x)\lambda(\mathrm{d}t)\\
&=\lim_{\tau_i\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\int_{B(\tau)}\left(\sum_{x\in\Omega}f(x)\exp\{it^{\mathrm{T}}x\}\right)\left(\sum_{x\in\Omega}f(x)\exp\{-it^{\mathrm{T}}x\}\right)\lambda(\mathrm{d}t)\\
&=\lim_{\tau_i\to\infty}\left(\prod_{i}1/(2\tau_i)\right)\int_{B(\tau)}\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\exp\{i(t^{\mathrm{T}}x-t^{\mathrm{T}}x')\}\lambda(\mathrm{d}t)\qquad\text{(Fubini-Tonelli)}\\
&=\lim_{\tau_i\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\int_{B(\tau)}\exp\{i(t^{\mathrm{T}}x-t^{\mathrm{T}}x')\}\lambda(\mathrm{d}t)\qquad\text{(Fubini-Tonelli)}\\
&=\lim_{\tau_i\to\infty}\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\left[\int_{B(\tau)}\exp\{i(t^{\mathrm{T}}x-t^{\mathrm{T}}x')\}\lambda(\mathrm{d}t)\right]\\
&=\lim_{\tau_i\to\infty}\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\left(\prod_{i=1}^{d}\left[1/(2\tau_i)\int_{-\tau_i}^{\tau_i}\exp\{i(t_i(x_i-x_i')\}\mathrm{d}t_i\right]\right)\qquad\text{(Fubini-Tonelli)}\\
&=\lim_{\tau_i\to\infty}\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\left(\prod_{i=1}^{d}\chi(x_i,x_i',\tau_i)\right)\quad\text{where}\quad\chi=\begin{cases}\frac{\sin\tau_i(x_i-x_i')}{\tau_i(x_i-x_i')}&\text{if } x_i\neq x_i',\\1&\text{else}\end{cases}\\
&=\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')\left(\lim_{\tau_i\to\infty}\prod_{i=1}^{d}\chi(x_i,x_i',\tau_i)\right)\qquad\text{(DCT)}\\
&=\sum_{x\in\Omega}\sum_{x'\in\Omega}f(x)f(x')1[x=x']\quad\text{where}\quad 1[\arg]=\begin{cases}1&\text{if arg holds,}\\0&\text{else}\end{cases}\\
&=\sum_{x\in\Omega}f^2(x).\end{aligned} \tag{26}$$

In details: the first equality follows by definition; the second and third by Fubini-Tonelli Theorem;[10] the fourth by simple rules of arithmetic; the fifth again by Fubini-Tonelli Theorem to decompose the volume calculation into a product; the sixth by evaluating the integral; seventh by the dominated convergence theorem;[11] the eighth by evaluating the limit; and the last by simple arithmetic. □

B.1.3 The Energy of Discrete Distributions as Described by their Characteristic Functions

Lemma B.2. *For any independent, discrete random variables A and B as described, taking values in $\mathbb{R}^d$,*

$$\varepsilon_{01}(A,B) = \lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\dots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d} 1/(2\tau_i)\right)\int_{B(\tau)} |\hat{p}_\alpha(t)-\hat{p}_\beta(t)|^2\lambda(\mathrm{d}t). \tag{27}$$

Proof. According to Székely and Rizzo (2013), for independent A and B, we have

$$\begin{aligned}|\hat{p}_\alpha(t)-\hat{p}_\beta(t)|^2 &= \mathbf{E}[\cos\{t^{\mathrm{T}}(A-A')\}+\cos\{t^{\mathrm{T}}(B-B')\}-\cos\{t^{\mathrm{T}}(A-B)\}]\\ &= \mathbf{E}\{2[1-\cos\{t^{\mathrm{T}}(A-B)\}]-[1-\cos\{t^{\mathrm{T}}(A-A')\}]-[1-\cos\{t^{\mathrm{T}}(B-B')\}]\}\end{aligned} \tag{28}$$

where A' and B' are i.i.d. copies of A and B, respectively. With the equivalence above, by Fubini's Theorem, we may interchange the expectation and integral in Eq. (27). We may also change the order of integration to arrive at

$$\begin{aligned}&\lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\dots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d} 1/(2\tau_i)\right)\int_{B(\tau)} |\hat{p}_\alpha(t)-\hat{p}_\beta(t)|^2\lambda(\mathrm{d}t)\\ &= \lim_{\tau_i\to\infty}\mathbf{E}\Bigg[\left(\prod_{i=1}^{d}\frac{1}{(2\tau_i)}\right)\int_{-\tau_1}^{\tau_1}\dots\int_{-\tau_d}^{\tau_d}\Big\{2\Big(1-\cos\sum_{i=1}^{d}\tau_i(A_i-B_i)\Big)\\ &\qquad -\Big(1-\cos\sum_{i=1}^{d}\tau_i(A_i-A_i')\Big)-\Big(1-\cos\sum_{i=1}^{d}\tau_i(B_i-B_i')\Big)\Big\}\mathrm{d}\tau_d\dots\mathrm{d}\tau_1\Bigg].\end{aligned} \tag{29}$$

To evaluate the integral we first observe, for any $x\in\mathbb{R}^d$,

$$\begin{aligned}\int_{-\tau_d}^{\tau_d} 1-\cos\sum_{i=1}^{d}\tau_i x_i\mathrm{d}\tau_d &= 2\tau_d - \frac{\sin\left(\tau_d x_d+\sum_{i=1}^{d-1}\tau_i x_i\right)-\sin\left(-\tau_d x_d+\sum_{i=1}^{d-1}\tau_i x_i\right)}{x_d}\\ &= 2\tau_d - \frac{2\cos\left(\sum_{i=1}^{d-1}\tau_i x_i\right)\sin(\tau_d x_d)}{x_d}.\end{aligned} \tag{30}$$

Notice, the above equation implies an iterative pattern which can be used to solve the multiple integral.

[10] The primary assumption of Fubini-Tonelli Theorem requires the *absolute value* of the integrand have finite double or iterated integral/sum. In the first case, with the iterated sum, it is clear for each fixed t since $\sum_x f(x)$ is bounded and so is $\exp\{-iz\}$ for all z. In the second and third cases, we simply cite the boundedness of $B(\tau)$ for each fixed τ.

[11] The primary assumption of the DCT is that the sequence of functions being integrated (or summed in our case) is dominated by some function g with finite integral (i.e., in the sense that the absolute value of every function in the sequence is less than or equal to g on all inputs). Again, this is easy to see using properties assumed on f and the fact that $|\chi|\leq 1$ for all inputs.

Keeping in mind which terms are constants with respect to the differential, we have

$$
\begin{aligned}
&\int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_{d-1}}^{\tau_{d-1}} \Big(\int_{-\tau_d}^{\tau_d} 1 - \cos \sum_{i=1}^{d} \tau_i x_i \mathrm{d}\tau_d \Big) \mathrm{d}\tau_{d-1} \dots \mathrm{d}\tau_1 \\
&= \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_{d-2}}^{\tau_{d-2}} \left(\int_{-\tau_{d-1}}^{\tau_{d-1}} 2\tau_d - \frac{2\cos\left(\sum_{i=1}^{d-1} \tau_i x_i\right) \sin(\tau_d x_d)}{x_d} \mathrm{d}\tau_{d-1} \right) \mathrm{d}\tau_{d-2} \dots \mathrm{d}\tau_1 \\
&= \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_{d-2}}^{\tau_{d-2}} \left((2\tau_d)(2\tau_{d-1}) - \frac{4\cos\left(\sum_{i=1}^{d-2} \tau_i x_i\right) \sin(\tau_d x_d) \sin(\tau_{d-1} x_{d-1})}{x_d x_{d-1}} \right) \mathrm{d}\tau_{d-2} \dots \mathrm{d}\tau_1 \\
&= \dots \\
&= \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_{d-j}}^{\tau_{d-j}} \left(\prod_{i=1}^{j} (2\tau_{d-i+1}) - \frac{\cos\left(\sum_{i=1}^{d-j} \tau_i x_i\right) \prod_{i=1}^{j} 2\sin(\tau_{d-i+1} x_{d-i+1})}{\prod_{i=1}^{j} x_{d-i+1}} \right) \mathrm{d}\tau_{d-j} \dots \mathrm{d}\tau_1 \\
&\dots \\
&= \prod_{i=1}^{d} (2\tau_{d-i+1}) - \frac{\prod_{i=1}^{d} 2\sin(\tau_{d-i+1} x_{d-i+1})}{\prod_{i=1}^{d} x_{d-i+1}} \\
&= \prod_{i=1}^{d} (2\tau_i) - \frac{\prod_{i=1}^{d} 2\sin(\tau_i x_i)}{\prod_{i=1}^{d} x_i}.
\end{aligned}
\tag{31}
$$

Now, returning to the RHS of Eq. (29), linearity of the integral implies

$$
\begin{aligned}
&\left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_d}^{\tau_d} \Big\{ 2\Big(1 - \cos \sum_{i=1}^{d} \tau_i (A_i - B_i)\Big) \\
&\qquad\qquad - \Big(1 - \cos \sum_{i=1}^{d} \tau_i (A_i - A_i')\Big) - \Big(1 - \cos \sum_{i=1}^{d} \tau_i (B_i - B_i')\Big) \Big\} \mathrm{d}\tau_d \dots \mathrm{d}\tau_1 \\
&= \left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_d}^{\tau_d} \Big\{ 2\Big(1 - \cos \sum_{i=1}^{d} \tau_i (A_i - B_i)\Big) \Big\} \mathrm{d}\tau_d \dots \mathrm{d}\tau_1 \\
&\quad - \left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_d}^{\tau_d} \Big\{ \Big(1 - \cos \sum_{i=1}^{d} \tau_i (A_i - A_i')\Big) \Big\} \mathrm{d}\tau_d \dots \mathrm{d}\tau_1 \\
&\quad - \left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_d}^{\tau_d} \Big\{ \Big(1 - \cos \sum_{i=1}^{d} \tau_i (B_i - B_i')\Big) \Big\} \mathrm{d}\tau_d \dots \mathrm{d}\tau_1.
\end{aligned}
\tag{32}
$$

Thus, we can apply the solution in Eq. (31) to solve the integral in Eq. (29). Taking $x_i = (A_i - B_i)$ in Eq. (31), we consider the first integral of Eq. (32) above along with its multiplicative constant:

$$
\begin{aligned}
&\left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \int_{-\tau_1}^{\tau_1} \dots \int_{-\tau_d}^{\tau_d} \Big(1 - \cos \sum_{i=1}^{d} \tau_i (A_i - B_i)\Big) \\
&= \left(\prod_{i=1}^{d} \frac{1}{(2\tau_i)} \right) \left(\prod_{i=1}^{d} (2\tau_i) - \frac{\prod_{i=1}^{d} 2\sin\Big\{\tau_i (A_i - B_i)\Big\}}{\prod_{i=1}^{d} (A_i - B_i)} \right) \\
&= 1 - \prod_{i=1}^{d} \frac{\sin\Big\{\tau_i (A_i - B_i)\Big\}}{\tau_i (A_i - B_i)} = 1 - \prod_{i=1}^{d} \chi(A_i, B_i, \tau_i)
\end{aligned}
\tag{33}
$$

where χ is defined in the proof of Eq. (23) (Lemma B.1). Taking $x_i = (A_i - A_i')$ and $x_i = (B_i - B_i')$ and proceeding as above allows us to resolve the entire integral. In particular, we have

$$\begin{aligned}
&\lim_{\tau_1\to\infty}\lim_{\tau_2\to\infty}\cdots\lim_{\tau_d\to\infty}\left(\prod_{i=1}^{d}1/(2\tau_i)\right)\int_{B(\tau)}|\hat{p}_\alpha(t)-\hat{p}_\beta(t)|^2\lambda(\mathrm{d}t)\\
&=\lim_{\tau_i}\mathbf{E}\left[2\Big(1-\prod_{i=1}^{d}\chi(A_i,B_i,\tau_i)\Big)-\Big(1-\prod_{i=1}^{d}\chi(A_i,A_i',\tau_i)\Big)-\Big(1-\prod_{i=1}^{d}\chi(B_i,B_i',\tau_i\Big)\right]\\
&=\mathbf{E}\left[\lim_{\tau_i}\left\{2\Big(1-\prod_{i=1}^{d}\chi(A_i,B_i,\tau_i)\Big)-\Big(1-\prod_{i=1}^{d}\chi(A_i,A_i',\tau_i)\Big)-\Big(1-\prod_{i=1}^{d}\chi(B_i,B_i',\tau_i\Big)\right\}\right]\\
&=\mathbf{E}\big[2\times 1[A_i\neq B_i]-1[A_i\neq A_i']-1[B_i\neq B_i']\big].
\end{aligned}\tag{34}$$

Here, the second equality follows from the dominated convergence theorem and $1[\arg]$ is defined as in proof of Eq. (23) (Lemma B.1). □

B.1.4 Moving from Real-Valued Discrete Variables to Any Discrete Variables

Lemma B.3. *Let $\tilde{A}$ and $\tilde{B}$ be any independent, discrete random variables over a countable set Ω (i.e., not necessarily contained in $\mathbb{R}^d$). Then,*

$$\sum_{x\in\Omega}|\tilde{p}_\alpha(x)-\tilde{p}_\beta(x)|=\varepsilon_{01}(\tilde{A},\tilde{B}).\tag{35}$$

where $\tilde{p}_\alpha$ and $\tilde{p}_\beta$ are the mass functions of $\tilde{A}$ and $\tilde{B}$, respectively.

Proof. Let $\Pi\subset\mathbb{R}^d$ with $|\Pi|=|\Omega|$. Note, Π exists because Ω is countable and $\mathbb{R}^d$ is not. Next, let $f:\Omega\to\Pi$ be any bijective map.

Then, supposing p_α and p_β are the mass functions of $f(\tilde{A})$ and $f(\tilde{B})$ respectively, by definition of the pushforward measure, for any $y\in\Pi$ such that $y=f(x)$ for $x\in\Omega$

$$p_\alpha(y)=\tilde{p}_\alpha(\{a\in\Omega\mid f(a)=y\})=\tilde{p}_\alpha(x).\tag{36}$$

Notice, bijectivity of f ensures the last step, because each $y\in\Pi$ has a *unique* inverse $x\in\Omega$. From bijectivity of f, we also have injectivity, which implies $1[a\neq b]=1[f(a)\neq f(b)]$ for all $a,b\in\Omega$. By simple substitution, the previous two facts tells us

$$\begin{aligned}
&2\sum_{a,b\in\Omega}1[a\neq b]\tilde{p}_\alpha(a)\tilde{p}_\beta(b)-\sum_{a,a'\in\Omega}1[a\neq a']\tilde{p}_\alpha(a)\tilde{p}_\alpha(a')-\sum_{b,b'\in\Omega}1[b\neq b']\tilde{p}_\beta(b)\tilde{p}_\beta(b')\\
&=2\sum_{a,b\in\Omega}1[f(a)\neq f(b)]p_\alpha(f(a))p_\beta(f(b))-\sum_{a,a'\in\Omega}1[f(a)\neq f(a)']p_\alpha(f(a))p_\alpha(f(a'))\\
&\quad-\sum_{b,b'\in\Omega}1[f(b)\neq f(b')]p_\beta(f(b))p_\beta(f(b'))
\end{aligned}\tag{37}$$

Since f is surjective too (i.e., along with injective), summation of any function $g(f(a),f(b))$ over $a,b\in\Omega$ and summation of $g(c,d)$ over $c,d\in\Pi$ are equivalent.[12] So, we can continue as follows:

$$\begin{aligned}
&2\sum_{a,b\in\Omega}1[f(a)\neq f(b)]p_\alpha(f(a))p_\beta(f(b))-\sum_{a,a'\in\Omega}1[f(a)\neq f(a)']p_\alpha(f(a))p_\alpha(f(a'))\\
&\quad-\sum_{b,b'\in\Omega}1[f(b)\neq f(b')]p_\beta(f(b))p_\beta(f(b'))\\
&=2\sum_{c,d\in\Pi}1[c\neq d]p_\alpha(c)p_\beta(d)-\sum_{c,c'\in\Omega}1[c\neq c']p_\alpha(c)p_\alpha(c')-\sum_{d,d'\in\Omega}1[d\neq d']p_\beta(d)p_\beta(d')
\end{aligned}\tag{38}$$

[12] In particular, because f is surjective, we know all pairs $(c,d)\in\Pi^2$ have some pair $(a,b)\in\Omega^2$ for which $(f(a),f(b))=(c,d)$; i.e., we do not "miss" a term in this sum. Because f is injective, we know all pairs $(c,d)\in\Pi^2$ have *only one* pair $(a,b)\in\Omega^2$ for which $(f(a),f(b))=(c,d)$; i.e., we do not "repeat" a term in this sum.

In other words, the previous two equations tell us $\varepsilon_{01}(\tilde{A}, \tilde{B}) = \varepsilon_{01}(f(\tilde{A}), f(\tilde{B}))$. Applying equivalence of the mass functions, then Lemmas B.1 and B.2, then equivalence of the energies:

$$\sum_{x\in\Omega} |\tilde{p}_\alpha(x) - \tilde{p}_\beta(x)| = \sum_{y\in\Pi} |p_\alpha(y) - p_\beta(y)| = \varepsilon_{01}(f(\tilde{A}), f(\tilde{B})) = \varepsilon_{01}(\tilde{A}, \tilde{B}). \tag{39}$$

Note, this uses the fact that functions of independent random variables are also independent. □

B.1.5 The Main Bound

Theorem B.1. *Let A and B be any independent random variables over any space $\mathcal{X}$ and let S, S' be random variables over $[0,1]$. Let U be a random variable, independent from A and B, over any set $\mathcal{U}$. Suppose $c : \mathcal{X} \to \Omega$ is a coarsening function (so, $\Omega \subset \mathcal{X}$) and let $f \in [0,1]^{\mathcal{X}\times\mathcal{U}}$. Then,*

$$\mathbf{E}[|S - f(A,U)|] \le \gamma + \varphi + \mathbf{E}[|S' - f(B,U)|] + \sqrt{\varepsilon_c(A,B) \times \delta} \tag{40}$$

where

$$\begin{aligned}
\gamma &= \mathbf{E}[|f(c(B),U) - f(B)|] + \mathbf{E}[|f(c(A),U) - f(A)|],\\
g &\in \operatorname*{arg\,min}_{h\in[0,1]^{\mathcal{X}\times\mathcal{U}}} \mathbf{E}[|S - h(c(A),U)|] + \mathbf{E}[|h(c(B),U) - S'|],\\
\varphi &= \mathbf{E}[|S - g(c(A),U)|] + \mathbf{E}[|g(c(B),U) - S'|],\\
\delta &= \sum_{x\in\Omega} |g(x) - f(x)|^2
\end{aligned} \tag{41}$$

Proof. For any $g \in [0,1]^{\mathcal{X}\times\mathcal{U}}$, by way of the triangle inequality and monotonicity of the expectation,

$$\begin{aligned}
&\mathbf{E}[|S - f(A,U)|] = \mathbf{E}[|S - f(A,U)|] + \mathbf{E}[|S' - f(B,U)|] - \mathbf{E}[|S' - f(B,U)|]\\
&= \mathbf{E}[|S - g(c(A),U) + g(c(A),U) - f(A,U)|] + \mathbf{E}[|S' - f(B,U)|] - \mathbf{E}[|S' - f(B,U)|]\\
&\le \mathbf{E}[|S - g(c(A),U)|] + \mathbf{E}[|g(c(A),U) - f(A,U)|] + \mathbf{E}[|S' - f(B,U)|]\\
&\qquad - \mathbf{E}[|S' - f(B,U)|]\\
&\le \mathbf{E}[|S - g(c(A),U)|] + \mathbf{E}[|g(c(A),U) - f(A,U)|] + \mathbf{E}[|S' - f(B,U)|]\\
&\qquad - \mathbf{E}[|g(c(B),U) - f(B,U)|] + \mathbf{E}[|g(c(B),U) - S'|]\\
&\le \mathbf{E}[|S - g(c(A),U)|] + \mathbf{E}[|g(c(A),U) - f(c(A),U)|] + \mathbf{E}[|f(c(A),U) - f(A,U)|]\\
&\qquad + \mathbf{E}[|S' - f(B,U)|] - \mathbf{E}[|g(c(B),U) - f(B,U)|] + \mathbf{E}[|g(c(B),U) - S'|]\\
&\le \mathbf{E}[|S - g(c(A),U)|] + \mathbf{E}[|g(c(A),U) - f(c(A),U)|] + \mathbf{E}[|f(c(A),U) - f(A,U)|]\\
&\qquad + \mathbf{E}[|S' - f(B,U)|] - \mathbf{E}[|g(c(B,U) - f(c(B),U)|]\\
&\qquad + \mathbf{E}[|f(c(B),U) - f(B,U)|] + \mathbf{E}[|g(c(B),U) - S'|].
\end{aligned} \tag{42}$$

Set $\tilde{B} = c(B)$, $\tilde{A} = c(A)$ and set

$$\begin{aligned}
\gamma &= \mathbf{E}[|f(\tilde{B},U) - f(B,U)|] + \mathbf{E}[|f(\tilde{A},U) - f(A,U)|],\\
g &\in \operatorname*{arg\,min}_{h\in[0,1]^{\mathcal{X}\times\mathcal{U}}} \mathbf{E}[|S - h(\tilde{A},U)|] + \mathbf{E}[|h(\tilde{B},U) - S'|],\\
\varphi &= \mathbf{E}[|S - g(\tilde{A},U)|] + \mathbf{E}[|g(\tilde{B},U) - S'|].
\end{aligned} \tag{43}$$

Then, Eq. (42) implies

$$\mathbf{E}[|S-f(A,U)|] \le \gamma+\varphi+\mathbf{E}[|S'-f(B,U)|]+\mathbf{E}[|g(\tilde{A},U)-f(\tilde{A},U)|]-\mathbf{E}[|g(\tilde{B},U)-f(\tilde{B},U)|]. \tag{44}$$

Now, suppose $\tilde{p}_\alpha$ and $\tilde{p}_\beta$ are probability mass functions for $\tilde{A}$ and $\tilde{B}$, respectively. Then, using basic properties of the expectation along with other noted facts,

$$\begin{aligned}
&\mathbf{E}[|g(\tilde{A},U)-f(\tilde{A},U)|]-\mathbf{E}[|g(\tilde{B},U)-f(\tilde{B},U)|] \\
&=\mathbf{E}\Big[\sum_{a\in\Omega}|g(a,U)-f(a,U)|\tilde{p}_\alpha(a)-\sum_{b\in\Omega}|g(b,U)-f(b,U)|\tilde{p}_\beta(b)\Big] \quad \text{(Fubini)} \\
&=\mathbf{E}\Big[\sum_{x\in\Omega}|g(x,U)-f(x,U)|(\tilde{p}_\alpha(x)-\tilde{p}_\beta(x))\Big]\leq\mathbf{E}\Big[\sum_{x\in\Omega}|g(x,U)-f(x,U)||\tilde{p}_\alpha(x)-\tilde{p}_\beta(x)|\Big] \\
&\leq\mathbf{E}\left[\left(\sum_{x\in\Omega}|g(x,U)-f(x,U)|^2\right)^{1/2}\left(\sum_{x\in\Omega}|\tilde{p}_\alpha(x)-\tilde{p}_\beta(x)|^2\right)^{1/2}\right] \quad \text{(Cauchy-Schwarz)} \\
&\leq\sqrt{\varepsilon_{01}(\tilde{A},\tilde{B})}\times\mathbf{E}\left[\left(\sum_{x\in\Omega}|g(x,U)-f(x,U)|^2\right)^{1/2}\right] \quad \text{(Lemma B.3)}
\end{aligned} \tag{45}$$

In the last step, we may apply Lemma B.3 because $\tilde{A}$ and $\tilde{B}$ are still independent (i.e., they are functions of independent random variables) and are now discrete too. Defining δ appropriately yields the result. □

B.1.6 Proof of Thm. A.1 and Other Applications of Thm. B.1

Thm. A.1 Thm. A.1 is simply a specification of Thm. B.1 above. In fact, it is better stated as a corollary of Thm. B.1. We set $\mathcal{X}=\mathcal{D}$, leave $\mathcal{U}$ and its variable U unchanged, and set $S=S'=h_\ell(D,U)$. Then, $A=\tilde{D}_1$ and $B=\tilde{D}_2$. Taking $f=h_\ell$ yields the result.

Classification and Regression In adaptation for classification and regression, we consider a source distribution $\mathbb{S}$ governing random variables (X_S,Y_S) and a target distribution $\mathbb{T}$ governing random variables (X_T,Y_T). In general, the goal is to predict $Y_\square$ from $X_\square$. We can set $S=Y_T$ and $S'=Y_S$. We may also set $A=X_T$ and $B=X_S$. Then, we learn f from a pre-specified *hypothesis class* $\mathcal{H}\subseteq[0,1]^{\mathcal{X}\times\mathcal{U}}$. Typically, U is ignored in these settings, but it seems possible to employ this term to model stochastic (Gibbs) predictors; i.e., in PAC-Bayesian Frameworks (Germain et al., 2020; Sicilia et al., 2022a). Notice, for regression, our framework only considers a normalized response variable and the mean absolute error.

B.1.7 Sample Complexity

As alluded in Section 6, a key shortcoming of our framework compared to existing frameworks is the absence of any terms measuring *sample-complexity*. That is, we do not explicitly quantify the difference between our empirical observation of the energy and the *true* energy (i.e., the *population* version of the statistic) using the number of samples in our observation. This is a big part of computational learning theory, as the act of choosing a function f *using data* – or, in dialogue contexts, choosing the parameter θ using data – can have significant impact on the difference between our observations of a statistical processes and reality. In fact, this impact is the basis of overfitting and, besides computational efficiency, is the main pillar of study in traditional PAC learning[13] (Valiant, 1984; Shalev-Shwartz and Ben-David, 2014). In more recent studies of domain adaptation, like our work, the population-only bound can be just as important for purpose of understanding and interpretation. Furthermore, if we only care about the empirical samples in-hand, these population-only bounds are directly applicable,[14] which partly explains the empirical effectiveness of our theory in Section 5. Nonetheless, the role of sample-complexity can be very informative and useful in practice (Pérez-Ortiz et al., 2021) and would be important for model-selection applications as described at the end of Appendix A. We leave investigation of sample-complexity as future work. As we are aware, there is currently no appropriate description of sample-complexity for dialogue generation contexts.

[13] Probably Approximately Correct learning

[14] The empirical sample becomes the whole population about which we are concerned.

C Statistics on Dataset

unique images	unique objects	words (+1 occurrences)	words (+3 occurrences)	questions
67K	134K	19K	6.6K	277K

Table 2: Statistics on *GuessWhat?!*. For more information (e.g., train/test splits) see original proposal (De Vries et al., 2017).

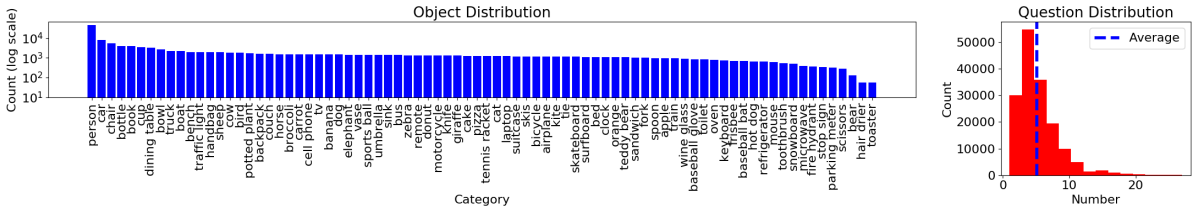

Figure 4: Visualization of object counts and dialogue length in *GuessWhat?!* dataset.

References

Katherine Atwell, Anthony Sicilia, Seong Jae Hwang, and Malihe Alikhani. 2022. The change that matters in discourse parsing: Estimating the impact of domain shift on parser error. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 824–845.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010a. A theory of learning from different domains. *Machine learning*, 79(1):151–175.

Shai Ben-David, Tyler Lu, Teresa Luu, and Dávid Pál. 2010b. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136. JMLR Workshop and Conference Proceedings.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Elia Bruni and Raquel Fernandez. 2017. Adversarial evaluation for open-domain dialogue generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 284–288.

R Cuppens. 1975. Decomposition of multivariate distributions.

Abhishek Das, Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. 2017. Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE international conference on computer vision*, pages 2951–2960.

Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5503–5512.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. 2020. Pac-bayes and domain adaptation. *Neurocomputing*, 379:379–397.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Mert Inan, Piyush Sharma, Baber Khalid, Radu Soricut, Matthew Stone, and Malihe Alikhani. 2021. Cosmic: A coherence-aware generation metric for image descriptions. *arXiv preprint arXiv:2109.05281*.

Fredrik D Johansson, David Sontag, and Rajesh Ranganath. 2019. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 527–536. PMLR.

Sham Machandranath Kakade. 2003. *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom).

Seiichi Kuroki, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato, and Masashi Sugiyama. 2019. Unsupervised domain adaptation based on source-guided discrepancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4122–4129.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. 2018. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

María Pérez-Ortiz, Omar Rivasplata, John Shawe-Taylor, and Csaba Szepesvári. 2021. Tighter risk certificates for neural networks. *Journal of Machine Learning Research*, 22.

Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. 2019. Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems*, 32.

Ievgen Redko, Amaury Habrard, and Marc Sebban. 2017. Theoretical analysis of domain adaptation with optimal transport. In *ECML PKDD*, pages 737–753. Springer.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. 2020. A survey on domain adaptation theory. *ArXiv*, abs/2004.11829.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Ravi Shekhar, Aashish Venkatesh, Tim Baumgärtner, Elia Bruni, Barbara Plank, Raffaella Bernardi, and Raquel Fernández. 2019. Beyond task success: A closer look at jointly learning to see, ask, and GuessWhat. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2578–2587, Minneapolis, Minnesota. Association for Computational Linguistics.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. 2018. Wasserstein distance guided representation learning for domain adaptation. In *AAAI*.

Anthony Sicilia, Katherine Atwell, Malihe Alikhani, and Seong Jae Hwang. 2022a. Pac-bayesian domain adaptation bounds for multiclass learners. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

Anthony Sicilia, Tristan Maidment, Pat Healy, and Malihe Alikhani. 2022b. Modeling non-cooperative dialogue: Theoretical and empirical insights. *Transactions of the Association for Computational Linguistics*, 10:1084–1102.

Florian Strub, Harm De Vries, Jeremie Mary, Bilal Piot, Aaron Courvile, and Olivier Pietquin. 2017. End-to-end optimization of goal-driven and visually grounded dialogue systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 2765–2771.

Gabor J Szekely. 1989. Potential and kinetic energy in statistics. *Lecture Notes, Budapest Institute*.

Gábor J Székely and Maria L Rizzo. 2013. Energy statistics: A class of statistics based on distances. *Journal of statistical planning and inference*, 143(8):1249–1272.

Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J Gordon. 2020. Domain adaptation with conditional distribution matching and generalized label shift. *Advances in Neural Information Processing Systems*, 33:19276–19289.

Leslie G Valiant. 1984. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019a. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019b. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR.

Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. 2019. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pages 7523–7532. PMLR.