2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC) | 978-1-6654-2851-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/SPAWC51858.2021.9593264

Single-Shot Compression for Hypothesis Testing

Fabrizio Carpi, Siddharth Garg, Elza Erkip
Department of Electrical and Computer Engineering, New York University, Brooklyn, NY
{fabrizio.carpi, siddharth.garg, elza}@nyu.edu

***Abstract*—Enhanced processing power in the cloud allows constrained devices to offload costly computations: for instance, complex data analytics tasks can be computed by remote servers. Remote execution calls for a new compression paradigm that optimizes performance on the analytics task within a rate constraint, instead of the traditional rate-distortion framework which focuses on source reconstruction. This paper considers a simple binary hypothesis testing scenario where the resource constrained client (transmitter) performs fixed-length single-shot compression on data sampled from one of two distributions; the server (receiver) performs a hypothesis test on multiple received samples to determine the correct source distribution. To this end, the task-aware compression problem is formulated as finding the optimal source coder that maximizes the asymptotic error performance of the hypothesis test on the server side under a rate constraint. A new source coding strategy based on a greedy optimization procedure is proposed and it is shown that that the proposed compression scheme outperforms universal fixed-length single-shot coding scheme for a range of rate constraints.**

***Index Terms*—Task-aware compression, source coding, fixed-length, single-shot, hypothesis testing.**

I. Introduction

Access to higher bandwidth and lower latency wireless technology is accelerating the use of edge computing. In edge computing, a resource constrained *client*, a mobile phone or a sensor for example, outsources computations to a remote *server* over a wireless link. Typically, the computations involve decision and analytics tasks over the transmitted data: for instance, image classification, object detection or speech recognition. For efficient bandwidth usage, the client might seek to compress the source data before transmitting to the server. However, traditional compression (or source coding) schemes are optimized for source reconstruction, that is, the seek to minimize a distortion metric (e.g., mean squared error) between the transmitted and the received data. Nonetheless, distortion does not directly correspond to the receiver's goal in the edge computing scenario. In this case, the receiver's goal is to maximize performance on the analytics tasks of interest. This gives rise to the central question of this paper: how can we design *task-aware* source coding schemes which provide *effective* representations of the source data so as to successfully carry out the analytics task?

One answer to this question is to use a distortion metric that is tailored for common analytics tasks. Motivated by this idea, recent works [1], [2] have studied the rate-distortion tradeoffs for the logarithmic loss distortion measure, since log-loss is commonly used in the machine learning community in the context of classification tasks. However, even log-loss distortion measure is ultimately a proxy for the analytics task at hand. How much better could one do by tailoring the compression scheme for the exact analytics task?

In this paper, we investigate task-aware compression for a simple edge computing scenario. We select binary hypothesis testing as a candidate task since it is both commonplace and well understood mathematically. In binary hypothesis testing the source data is sampled from one of two distributions and the goal is to decide which one was the correct source distribution.

Next, we model the client's resource constraints — an unconstrained client could perform the hypothesis test by itself and transmit a single bit (binary decision) to the server. In contrast, our primary assumption is that the client does not have processing capabilities to compute the task locally. We model a resource-constrained client that only has sufficient resources to store and process a *single* data sample at a time; as such, it compresses each data sample it receives using a simple *scalar* compression scheme (as opposed to vector compression) and transmits to the server, over a rate-limited link. In literature, this is referred to as "single-shot" compression. We assume fixed-length (lossy) compression, i.e., the compressed samples belong to an alphabet with size limited by the rate constraint. The server, on the other hand, is computationally unconstrained and collects an arbitrarily large number of compressed samples from the client for hypothesis testing.

Versions of this problem have been investigated in a multi-terminal setting with compression over large blocklengths [3]. In most of this literature, no resource constraints are assumed on the clients and the asymptotic performance is provided. Ziv [4] analyzes binary hypothesis testing with empirically observed statistics; a link to universal compression is established but applies only over large blocklengths, while we are interested in single-shot compression. Prior work has also looked at the related problem of learning classification-oriented compressed data representations [5], where both the client and server operate on a single sample of data as it is customary in classification settings, as opposed to hypothesis testing that operates over large blocklengths [6].

The main focus of this paper is to design an effective task-aware source coder for binary hypothesis testing. In Section II, we start by formally defining the system model, where we take into account the client constraints mentioned above. In Section III, we formalize the fixed-length single-shot compression for hypothesis testing problem; we also

This work was supported in part by NSF–Intel grant #2003182 and NSF grant #1925079.

2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)

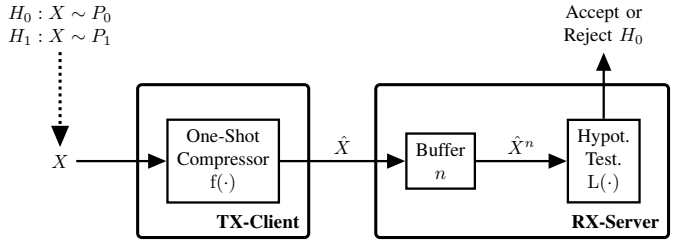

Fig. 1: System Model.

define the optimal compressor, which requires exponential (in the alphabet size) complexity for the construction. Then, we propose a task-oriented compression scheme in Section IV: our scheme is based on a greedy optimization which aims to the preserve the *useful* information between the two source hypotheses, in this case the Kullback-Leibler distance between the two distributions. The proposed compressor is constructed through iterative steps and it can be determined in polynomial time. In Section V, we show empirical results and computational bounds for our compressor. Finally, our conclusions are discussed in Section VI.

II. System Model

The system model is shown in Fig. 1. Throughout the rest of the paper the client is called transmitter and the server is called receiver. The data comes from one of the two distributions P_θ, $\theta \in \{0, 1\}$, where $\theta = 0$ represents the null hypothesis H_0 and $\theta = 1$ represents the alternative hypothesis H_1. We have $X_1, \ldots, X_n \sim P_\theta$ i.i.d. random variables defined over a finite alphabet $\mathcal{X} = \{1, \ldots, |\mathcal{X}|\}$. The transmitter, due to memory constraints, cannot store and process X^n jointly to do hypothesis testing. Instead, it sends the one-shot (scalar) compressed X^n to the receiver where hypothesis testing takes place.

Formally, at the transmitter, the single-shot compressor f is a surjective function defined as

$$\mathrm{f} : \mathcal{X} \to \mathcal{M} \tag{1}$$

where $\mathcal{M} = \{1, \ldots, M\}$ is the compressed alphabet of size M. We denote $\hat{X} = \mathrm{f}(X)$, i.e., $\hat{X}$ represents the mapping of the source letter X. We consider $M < |\mathcal{X}|$, since for $M \geq |\mathcal{X}|$ there is no need for compression. This corresponds to fixed rate compression with rate $R = \log M$.[1]

The probability distribution of $\hat{X}$ under P_θ, $\theta \in \{0, 1\}$, is denoted as $\hat{P}_\theta$ and is given by

$$\hat{P}_\theta(\hat{x}) = \sum_{x : \mathrm{f}(x) = \hat{x}} P_\theta(x). \tag{2}$$

The receiver observes $\hat{X}_1, \ldots, \hat{X}_n$ and either accepts or rejects the null hypothesis. Using standards definitions in simple hypothesis testing [7], type-I error, denoted as α_n, occurs when the null hypothesis ($\theta = 0$) is true, but the receiver rejects it. Instead, type-II error, denoted as β_n, corresponds to the receiver accepting the null hypothesis when the alternative hypothesis ($\theta = 1$) is true. It is known that in the classical hypothesis testing setting, for any $\epsilon \in (0, 1/2)$ and $\alpha_n < \epsilon$, the optimal type-II error β_n^ϵ decays exponentially in n with exponent γ defined as

$$\gamma = -\lim_{n \to \infty} \frac{1}{n} \log \beta_n^\epsilon. \tag{3}$$

We say that (R, η) is *achievable* if there exists a single-shot rate R compressor at the client and a corresponding hypothesis testing function at the server with type-I error less than ϵ and type-II error exponent η. Note that type-II error exponent does not typically depend on type-I error bound ϵ [7] as long as ϵ is fixed, hence we will not explicitly state the dependency on ϵ. In particular, for a given compression rate R, we would like to find the largest achievable type-II error exponent

$$\gamma^\star(R) = \sup\{\eta : (R, \eta) \text{ achievable}\}. \tag{4}$$

Note that if $R = \log(|\mathcal{X}|)$ and the compressor is the identity transformation $\mathrm{id}(\cdot)$, then Chernoff-Stein lemma [7] determines the optimal error exponent

$$\gamma^\star(\log|\mathcal{X}|) = \gamma_{\mathrm{id}}(\log|\mathcal{X}|) = D(P_0||P_1), \tag{5}$$

where $D(P_0||P_1)$ is the Kullback–Leibler (KL) divergence between P_0 and P_1 [7]. The error exponent penalty for a rate R compressor f at is defined as

$$\Delta_{\mathrm{f}}(R) = D(P_0||P_1) - \gamma_{\mathrm{f}}(R), \tag{6}$$

where $\gamma_{\mathrm{f}}(R)$ is the largest type-II error exponent determined by the compressor f. The optimal penalty is

$$\Delta^\star(R) = D(P_0||P_1) - \gamma^\star(R). \tag{7}$$

III. Hypothesis Testing under Single-Shot Compression

For the one-shot compressed binary hypothesis testing problem, our first result states that the log-likelihood ratio (LLR) test using the compressed variables $\hat{X}_1, \ldots, \hat{X}_n$ is optimal.

Lemma 1 (Hypothesis testing on compressed variables)**.** *The following LLR test on compressed variables $\hat{X}_i = \mathrm{f}(X_i)$, $i = 1, \ldots, n$, is optimal.*

$$L(\hat{X}_1, \ldots, \hat{X}_n) = \sum_{i=1}^{n} \log \frac{\hat{P}_0(\hat{X}_i)}{\hat{P}_1(\hat{X}_i)} \underset{\hat{\theta}=1}{\overset{\hat{\theta}=0}{\gtrless}} \log T, \tag{8}$$

where $T \geq 0$ depends on the type-I error exponent bound ϵ. The corresponding optimal error exponent is

$$\gamma_{\mathrm{f}}(R) = D(\hat{P}_0||\hat{P}_1). \tag{9}$$

Proof sketch. Since the source random variable is i.i.d. and the compressor function is f memoryless, the compressed variable is also i.i.d. $\hat{X}_1, \ldots, \hat{X}_n \sim \hat{P}_\theta$. Then, Neyman-Pearson test [7, Chapter 11] can be applied to $\hat{X}^n$. Moreover, Chernoff-Stein lemma determines that the the optimal error exponent is equal to the KL divergence between the distribution of the compressed variables under the two hypotheses. □

[1] Throughout this paper $\log(\cdot)$ is assumed to be base 2.

As discussed in Section II, the error exponent $\gamma_{\mathrm{f}}(R)$ determines the speed of convergence — intuitively, the farther apart the two compressed distributions (large KL divergence), the faster the type-II error probability goes to zero. Hence, our goal is to find a compressor f which induces a partition of M sets over $\mathcal{X}$ such that the KL distances between the compressed distributions $D(\hat{P}_0||\hat{P}_1)$ is maximized. Clearly, compression reduces the error exponent (we will mathematically show this in Proposition 1) and by Lemma 1 the smallest compression penalty for the compressor f is

$$\Delta_{\mathrm{f}}(R) = D(P_0||P_1) - D(\hat{P}_0||\hat{P}_1). \tag{10}$$

Then, the optimal compressor $\mathrm{f}^\star$ at rate $R = \log M$ is

$$\mathrm{f}^\star = \arg\max_{\mathrm{f}} D(\hat{P}_0||\hat{P}_1) \quad \text{s.t. } |\mathrm{f}| \leq M, \tag{11}$$

or, equivalently,

$$\mathrm{f}^\star = \arg\min_{\mathrm{f}} \Delta_{\mathrm{f}}(R) \quad \text{s.t. } |\mathrm{f}| \leq M. \tag{12}$$

where $|\mathrm{f}|$ is the cardinality of the compression function.

In the following proposition we derive a useful analytical expression for $\Delta_{\mathrm{f}}(R)$ in terms of distributions over compressed symbols. For mathematical convenience, we define $\mathcal{G}_{\hat{x}} = \{x : \mathrm{f}(x) = \hat{x}\}$; this set includes the source outcomes mapped to the compressed symbol $\hat{x}$. Hence, the compressor induces the "groups" $\mathcal{G}_{\hat{x}} \in \{\mathcal{G}_1, \ldots, \mathcal{G}_M\} = \mathcal{G}$, which form a partition over $\mathcal{X}$.

Proposition 1 (Compression penalty on type-II error exponent)**.** *For any compressor* f*, the minimal compression penalty is* $\Delta_{\mathrm{f}}(R) \geq 0$ *and can be expressed as:*

$$\Delta_{\mathrm{f}}(R) = \sum_{\hat{x}=1}^{M} \hat{P}_0(\hat{x})\, D\Big(P_0(x|\hat{x})\Big|\Big|P_1(x|\hat{x})\Big) \tag{13}$$

where the posterior distribution of X *given the compressed realization* $\mathrm{f}(X) = \hat{x}$ *is*

$$P_\theta(x|\hat{x}) = \frac{P_\theta(x)}{\hat{P}_\theta(\hat{x})} \mathbb{1}\{\hat{x} = \mathrm{f}(x)\}. \tag{14}$$

Proof. Expanding equation (10):

$$\begin{aligned}
\Delta_{\mathrm{f}}(R) &= \sum_{x\in\mathcal{X}} P_0(x) \log \frac{P_0(x)}{P_1(x)} - \sum_{\hat{x}\in\mathcal{M}} \hat{P}_0(\hat{x}) \log \frac{\hat{P}_0(\hat{x})}{\hat{P}_1(\hat{x})} \\
&= \sum_{\hat{x}\in\mathcal{M}} \sum_{x\in\mathcal{G}_{\hat{x}}} P_0(x) \log \frac{P_0(x)}{P_1(x)} - \sum_{\hat{x}\in\mathcal{M}} \left(\sum_{x\in\mathcal{G}_{\hat{x}}} P_0(x)\right) \log \frac{\hat{P}_0(\hat{x})}{\hat{P}_1(\hat{x})} && (15) \\
&= \sum_{\hat{x}\in\mathcal{M}} \sum_{x\in\mathcal{G}_{\hat{x}}} P_0(x) \log \left(\frac{P_0(x)}{\hat{P}_0(\hat{x})} \frac{\hat{P}_1(\hat{x})}{P_1(x)}\right) && (16) \\
&= \sum_{\hat{x}\in\mathcal{M}} \sum_{x\in\mathcal{G}_{\hat{x}}} P_0(x) \log \frac{P_0(x|\hat{x})}{P_1(x|\hat{x})} && (17) \\
&= \sum_{\hat{x}\in\mathcal{M}} \hat{P}_0(\hat{x})\, D\Big(P_0(x|\hat{x})\Big|\Big|P_1(x|\hat{x})\Big)
\end{aligned}$$

where: in (15) we used the definition (2); in (15) and (16) we used the fact that $\mathcal{G}_1, \ldots, \mathcal{G}_M$ form a partition over $\mathcal{X}$; in (17) we used the definition (14) since $P(\hat{X}|X) = \mathbb{1}\{\hat{X} = \mathrm{f}(X)\}$. Note that if $\mathcal{G}_{\hat{x}}$ contains a single element (one-to-one mapping), then $D\big(P_0(x|\hat{x})||P_1(x|\hat{x})\big) = 0$. Moreover (15) is greater than zero by the log-sum inequality. □

Algorithm 1: KL-greedy compressor's construction

Input : Source distributions P_0, P_1; rate M.
Initialize: $\hat{P}_0 \leftarrow P_0$, $\hat{P}_1 \leftarrow P_1$, $\mathcal{G} \leftarrow \{\{1\}, \ldots, \{|\mathcal{X}|\}\}$.
for $k = 1, \ldots, |\mathcal{X}| - M$ **do**
  Find $\{\mathcal{G}_a, \mathcal{G}_b\} \subset \mathcal{M}_k$ which minimize (18).
  Remove the b-th entry and combine $\{\mathcal{G}_a, \mathcal{G}_b\}$ by updating the a-th entry:
  $\hat{P}_0 \leftarrow [\ldots, \hat{P}_0(\mathcal{G}_a) + \hat{P}_0(\mathcal{G}_b), \ldots, 0, \ldots]$
  $\hat{P}_1 \leftarrow [\ldots, \hat{P}_1(\mathcal{G}_a) + \hat{P}_1(\mathcal{G}_b), \ldots, 0, \ldots]$
  $\mathcal{G} \leftarrow [\ldots, \mathcal{G}_a \cup \mathcal{G}_b, \ldots, \emptyset, \ldots]$
end
Output: Compressed distr. $\hat{P}_0, \hat{P}_1$; groups $\mathcal{G}$.

Non-negativity of $\Delta_{\mathrm{f}}(R) \geq 0$ can also be observed from equation (13) as it is a convex combination of KL-distances, each individually positive. Proposition 1 also yields an important intuition about optimal compression: note that the $\hat{x}$-th term in (13) is directly proportional to the relative entropy between the posteriors over the $\hat{x}$-th group $\mathcal{G}_{\hat{x}}$ induced by f. As a consequence, (13) suggests that a good task-aware compression strategy combines the source letters that have similar posteriors over the compressed groups; in other words, the probability ratios between the combined letters under P_0 has to be similar to the ones under P_1.

IV. Proposed Compressor

When solving the optimization problem in (11), one has to consider all the possible surjective functions f which induce valid partitions over the source alphabet; the number of such number of partitions is exponential in the source/compressed alphabet size. Partitioning problems of this nature have been shown to be NP-Hard [8, Chapter 3], [9].

In this paper, we propose an efficient (i.e., polynomial time) greedy approximation for the optimal compressor. The following lemma is the basis for our construction.

Lemma 2 (One-step Compression from $|\mathcal{X}|$ to $|\mathcal{X}| - 1$)**.** *Let* f *be a compression rule which groups two letters* $\{a, b\} \subset \mathcal{X}$*. That is,* $\mathcal{G}_m = \{a, b\}$*,* $m \in \mathcal{M}$*, and the others groups* $\mathcal{G}_i$*,* $i = 1, \ldots, M$*,* $i \neq m$*, are one-to-one. Then, the optimal compressor for* $M = |\mathcal{X}| - 1$ *induces the groups* $\mathcal{G}^\star$*, minimizing the compression penalty*

$$\mathcal{G}^\star = \arg\min_{\mathcal{G}_m = \{a,b\} \subset \mathcal{X}} \left\{ \hat{P}_0(m) D\Big(P_0(x|m)\Big|\Big|P_1(x|m)\Big) \right\}, \tag{18}$$

where the posteriors over the candidate group $\mathcal{G}_m = \{a, b\}$ *are simply defined as*

$$P_\theta(x|m) = \left[\frac{P_\theta(a)}{P_\theta(a) + P_\theta(b)}, \frac{P_\theta(b)}{P_\theta(a) + P_\theta(b)}\right]. \tag{19}$$

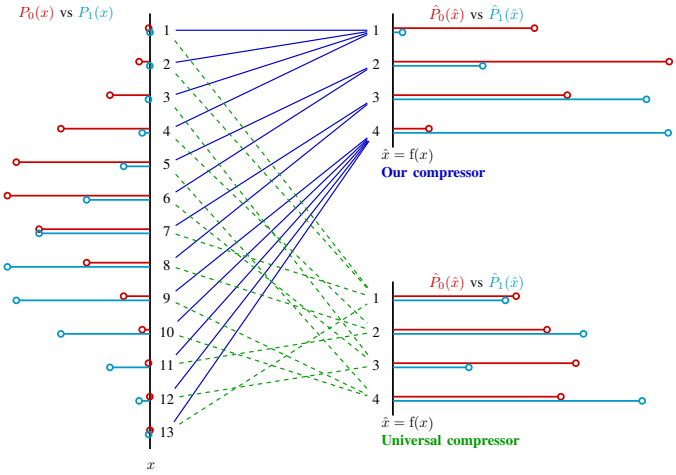

Fig. 2: Left: Source distributions for $|\mathcal{X}| = 13$. Top-right: compressed distributions for our compressor of Algorithm 1; the solid blue line shows the mappings of the compression function. Bottom-right: compressed distributions for the universal compressor from [2]; the dashed green line shows the mappings of the compression function.

Note that if the groups $\mathcal{G}_i$ are one-to-one, the i-th KL divergence term in (13) is 0. Intuitively, when reducing the alphabet size by one, the optimal compressor combines the two letters that minimize the product of the probability of the group and the KL distance between the posteriors over the group.

For general M, we propose an iterative construction of the compressor that reduces the compressed alphabet size by one in each step. Denote the steps by $k = 1, \ldots, |\mathcal{X}| - M$, where M is the target rate. Let $\mathcal{M}_k$ be the compressed alphabet at the k-th step, with size $|\mathcal{M}_k| = |\mathcal{X}| - k$, with $k = 1, \ldots, |\mathcal{X}| - M$. Let $\mathcal{G}_1, \ldots, \mathcal{G}_{|\mathcal{M}_k|}$ be the corresponding partition on $\mathcal{X}$ at the k-th step. For example, at the first step $k = 1$, the (optimal) groups $\mathcal{G}_1, \ldots, \mathcal{G}_{|\mathcal{X}|-1}$ are computed according to Lemma 2. Generally, at step $k > 1$, our compressor combines the two groups $\{\mathcal{G}_a, \mathcal{G}_b\}_k^\star \subset \mathcal{M}_k$ that minimize (18), where $\mathcal{X}$ is replaced by $\mathcal{M}_k$ and $\{\mathcal{G}_a, \mathcal{G}_b\}$ is a generalization of $\{a, b\}$. Finally, the compression function f is defined such that $\mathrm{f}(x) = \hat{x}$ if $x \in \mathcal{G}_{\hat{x}}$. We call our proposed compressor "KL-greedy" and its construction is summarized in Algorithm 1. Note that the number of pairs of groups $\{\mathcal{G}_a, \mathcal{G}_b\}$ that need to be considered at the k-th step is $\binom{|\mathcal{M}_k|}{2}$. Thus, our compressor can be designed in polynomial time.

V. Results

In this section, we discuss numerical results and performance of Algorithm 1. We consider P_θ to be a (shifted) binomial distribution over $\mathcal{X}$ with parameter s_θ, i.e.,

$$P_\theta(x) = \binom{|\mathcal{X}| - 1}{x - 1} s_\theta^{x-1} (1 - s_\theta)^{|\mathcal{X}| - x}. \tag{20}$$

We quantify the compression penalty $\Delta_{\mathrm{f}}(R)$ based on (10). We also estimate type-II error rate by performing the LLR test (8) on the receiver side; we consider blocklength $n = 5$ and bound on the type-I error $\epsilon = 0.05$. The threshold T is empirically chosen such that it is the largest value for which the estimated type-I error is $N(\hat{\theta} = 1, \theta = 0)/N(\theta = 0) < \epsilon$, for a given compressor f at rate M; $N(\cdot)$ is the counting function. The type-II error rate is empirically estimated as $N(\hat{\theta} = 0, \theta = 1)/N(\theta = 1)$. Both estimates are computed over $N(\theta = 0) = N(\theta = 1) = 10^6$ realizations of source blocks x^n.

A. Baseline: Single-shot Universal Lossy Source Coding under Logarithmic Loss

Universal compression schemes are designed to perform well over a family of source distributions — the family $\{P_0, P_1\}$ in our scenario. In compliance with our system model, we consider the universal fixed-length single-shot lossy compression scheme analyzed by Shkel et al. in [2]. We recall that although this universal compressor is task-unaware, it is designed for soft reconstruction under logarithmic loss distortion, which generally provides "universally good" schemes [10]. The construction of this universal compressor aims to find $Q^\star$, a distribution over $\mathcal{X}$ which is used to approximate the source distribution over the family $\{P_0, P_1\}$. As in [2], for a rate constraint $R = \log M$, $Q^\star$ belongs to

$$\mathcal{Q}_M = \{Q : \min_{x \in \mathcal{X}} \log \frac{1}{Q(x)} \geq \log M\}. \tag{21}$$

For every value of M, $Q^\star$ is the solution of the following optimization problem

$$Q^\star = \arg\min_{Q \in \mathcal{Q}_M} \delta \quad \text{s.t.:} \begin{cases} D(P_0||Q) \leq \delta, \\ D(P_1||Q) \leq \delta. \end{cases} \tag{22}$$

In other words, $Q^\star$ can be seen as a distribution that is "equidistant" from the two hypotheses. Given $Q^\star$, the universal compressor is constructed according to [1, Theorem 4]. Intuitively, the letters corresponding to the largest values of $Q^\star$ get one-to-one mappings, while the letters corresponding to the lowest values of $Q^\star$ get grouped together.

B. Simulation Results

We show the performance of different compressors in our hypothesis testing scenario. In the figures, we show empirical results for different compression functions f:

- Uncompressed: no compression is performed, i.e., $\hat{x} = x$;
- Optimal compressor: defined in (12);
- Our KL-greedy compressor: defined in Section IV and Algorithm 1;
- Universal compressor: defined in [2] and briefly introduced in Section V-A.

In Fig. 2, 3 and 4 we consider a source alphabet of size $|\mathcal{X}| = 13$; the parameters of the two hypotheses are $s_0 = 0.4$, $s_1 = 0.6$. On the other hand, in Fig. 5 and 6 we consider a larger source alphabet of size $|\mathcal{X}| = 256$; the parameters of the two hypotheses are $s_0 = 0.48$, $s_1 = 0.52$. We note that for this larger source alphabet, it is no longer computationally feasible to determine the optimal compressor.

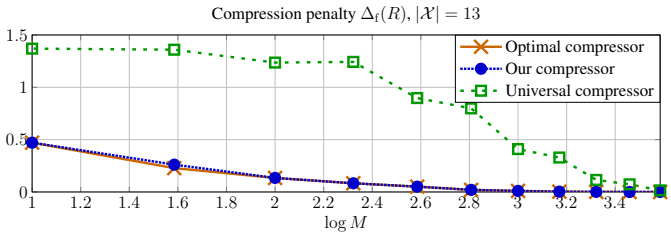

Fig. 3: Compression penalty for $|\mathcal{X}| = 13$.

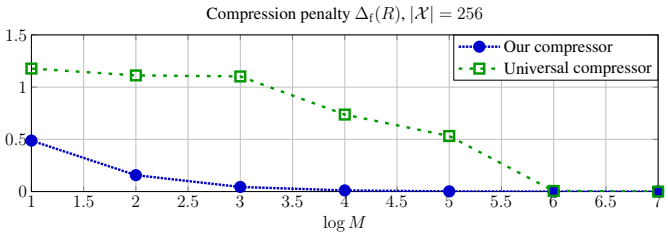

Fig. 5: Compression penalty for $|\mathcal{X}| = 256$.

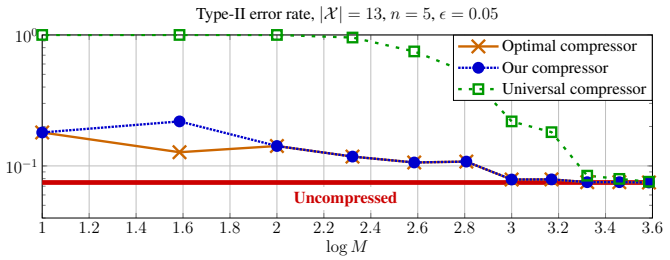

Fig. 4: Type-II error rates for $|\mathcal{X}| = 13$.

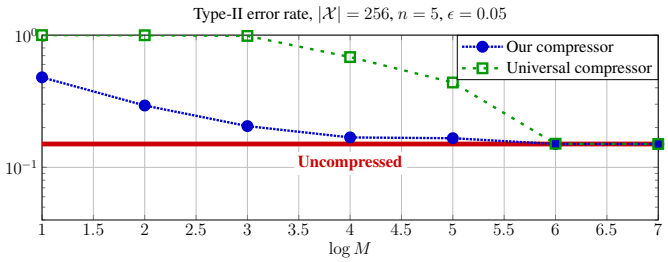

Fig. 6: Type-II error rates for $|\mathcal{X}| = 256$.

Fig. 2 illustrates the resulting KL-greedy compressor, the universal compressor, and the compressed distributions for $M = 4$. As discussed in Section III, our KL-greedy compressor seeks to minimize the KL distance between the posteriors over the groups; we also point out that this induces a partition on $\mathcal{X}$ that divides the source alphabet in regions where one of the hypothesis is more likely than the other. This pattern is also visible in the compressed distributions, since the two hypotheses exhibit divergent distributions (large KL distance). On the other hand, the universal compressor aims to make the two compressed distributions as uniform as possible. Clearly, as we discussed in Section III, the larger KL divergence between the compressed distributions, the better for the hypothesis testing task.

Fig. 3 and 5 show the compression penalty as a function of the compression rate M. The former also shows the performance of the optimal compressor, since it can be computed in reasonable time for a small source alphabet; in this case, we can see that our compressor performs close to the optimal. In both cases, our compressor outperforms the universal compressor, and it quickly achieves zero penalty, i.e., the KL distance of the compressed distributions is close to the uncompressed one as M increases.

Fig. 4 and 6 show the empirical type-II error rate as a function of the compression rate M. The former also shows the performance of the optimal compressor: our compressor performance overlaps with the optimal compressor. For both the small and the large alphabet scenarios, our compressor outperforms the universal compressor, and it quickly achieves an error rate close to the uncompressed setting as M increases.

VI. Conclusion

In this paper, we have analyzed one-shot lossy source coding for task-oriented communications. We have provided a problem formulation where the transmitter has to compress data coming from one of two distribution, and the goal is to carry out hypothesis testing at the receiver side. We have proposed a greedy compression function which can be determined in polynomial time and aims to preserve the *useful* information for hypothesis testing at the receiver. Namely, our scheme is designed to minimize the gap between the KL divergences at the source and after compression. Our experimental results show that our compressor outperforms classical universal compression schemes and achieves error rate comparable to the uncompressed case even for low rates.

References

[1] Y. Y. Shkel and S. Verdú, "A single-shot approach to lossy source coding under logarithmic loss," *IEEE Transactions on Information Theory*, vol. 64, no. 1, pp. 129–147, January 2018.

[2] Y. Shkel, M. Raginsky, and S. Verdú, "Universal lossy compression under logarithmic loss," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 1157–1161.

[3] T. S. Han and S. Amari, "Statistical inference under multiterminal data compression," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2300–2324, October 1998.

[4] J. Ziv, "On classification with empirically observed statistics and universal data compression," *IEEE Transactions on Information Theory*, vol. 34, no. 2, pp. 278–286, March 1988.

[5] C. T. Li, X. Wu, A. Özgür, and A. El Gamal, "Minimax learning for distributed inference," *IEEE Transactions on Information Theory*, vol. 66, no. 12, pp. 7929–7938, December 2020.

[6] J. J. Li and X. Tong, "Statistical hypothesis testing versus machine learning binary classification: Distinctions and guidelines," *Patterns*, vol. 1, no. 7, p. 100115, October 2020.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006.

[8] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman & Co., 1990.

[9] K. Wei, R. Iyer, S. Wang, W. Bai, and J. Bilmes, "Mixed robust/average submodular partitioning: Fast algorithms, guarantees, and applications," in *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'15. Cambridge, MA, USA: MIT Press, December 2015, p. 2233–2241.

[10] A. No, "Universality of logarithmic loss in fixed-length lossy compression," *Entropy*, vol. 21, no. 6, June 2019.