Semantic Information Recovery in Wireless Networks

Edgar Beck, *Graduate Student Member, IEEE,* Carsten Bockelmann, *Member, IEEE,* and Armin Dekorsy, *Senior Member, IEEE*

***Abstract*—Motivated by the recent success of Machine Learning (ML) tools in wireless communications, the idea of semantic communication by Weaver from 1949 has received considerable attention. It breaks with the classic design paradigm of Shannon by aiming to transmit the meaning of a message, i.e., semantics, rather than its exact copy and thus allows for savings in information rate. In this work, we extend the fundamental approach from Basu et al. for modeling semantics to the complete communications Markov chain. Thus, we model semantics by means of hidden random variables and define the semantic communication task as the data-reduced and reliable transmission of messages over a communication channel such that semantics is best preserved. We cast this task as an end-to-end Information Bottleneck problem allowing for compression while preserving relevant information at most. As a solution approach, we propose the ML-based semantic communication system SINFONY and use it for a distributed multipoint scenario: SINFONY communicates the meaning behind multiple messages that are observed at different senders to a single receiver for semantic recovery. We analyze SINFONY by processing images as message examples. Numerical results reveal a tremendous rate-normalized SNR shift up to 20 dB compared to classically designed communication systems.**

***Index Terms*—Semantic communication, wireless communications, wireless networks, infomax, information bottleneck, information theory, machine learning.**

I. Introduction

When Shannon laid the theoretical foundation of the research area of communications engineering back in 1948, he deliberately excluded semantic aspects from the system design [1], [2]. In fact, the idea of addressing semantics in communications arose shortly after Shannon's work in [2] but it remained largely unexplored. Since then, the design focus of communication systems has been on digital error-free symbol transmission. In recent years, it has become clear that semantics-agnostic communication limits the achievable efficiency in terms of bandwidth, power, and complexity trade-offs. Notable examples include wireless sensor networks and broadcast scenarios [3].

Owing to the great success of Artificial Intelligence (AI) and, in particular, its subdomain Machine Learning (ML), ML tools have been recently investigated for wireless communications [4], [5], [6]. Now, ML with its ability to extract features appears to be a proper means to realize a semantic design. Further, we note that the latter design is supported and possibly enabled by the 6G vision of integrating AI and ML on all layers of the communications system design, i.e., by an ML-native air interface.

This work was partly funded by the Federal State of Bremen and the University of Bremen as part of the Human on Mars Initiative.

The authors are with the Gauss-Olbers Space Technology Transfer Center c/o Department of Communications Engineering, University of Bremen, 28359 Bremen, Germany (e-mail: {beck, bockelmann, dekorsy}@ant.uni-bremen.de).

Motivated by these new ML tools and driven by high bandwidth and latency demands of the next wireless communication standard 6G, the idea of semantic communication has received considerable attention [2], [7], [8], [9], [10]. It breaks with the existing classic design paradigms by including semantics in the design of the wireless transmission. The goal of such a transmission is, therefore, to deliver the required data from which the highest levels of quality of information may be derived, as perceived by the application and/or the user. More precisely, semantic communication aims to transmit the meaning of a message rather than its exact copy and hence allows for compression and coding to the actual semantic content. Thus, savings in bandwidth, power, and complexity are expected.

A. Related Work

The notion of semantic communication traces back to Weaver [2] who reviewed Shannon's information theory [1] in 1949 and amended considerations w.r.t. semantic content of messages. Oftentimes quoted is his statement that *"there* ***seem*** *to be [communication] problems at three levels"* [2]:

A. How accurately can the symbols of communication be transmitted? (The technical problem.)
B. How precisely do the transmitted symbols convey the desired meaning? (The semantic problem.)
C. How effectively does the received meaning affect conduct in the desired way? (The effectiveness problem.)

Since then semantic communication was mainly investigated from a philosophical point of view, see, e.g., [11], [12].

In [13], [14], the authors extend the propositional logic-based approach from one of the earliest works [15] into a model-theoretical framework and define semantic information source and semantic channel. In particular, the authors consider a semantic source that *"observes the world and generates meaningful messages characterizing these observations"* [14]. The source is equivalent to conclusions, i.e., "models" of the world, that are unequivocally drawn following a set of known inference rules based on observation of messages. In [13], the authors consider joint semantic compression and channel coding at Level B with the classic transmission system, i.e., Level A, as the (semantic) channel. In contrast, [14] only deals with semantic compression and uses a different definition of the semantic channel (which we will make use of in this article): It is equal to the entailment relations between "models" and "messages". By this means, the authors are able to derive semantic counterparts of the source and channel coding theorems. However, as the authors admit, these theorems do not tell how to develop optimal coding algorithms and the assumption of a model-theoretical description leads to *"many non-trivial simplifications"* [13].

In [16], the authors define semantic similarity as a semantic error measure to quantify the distance between the meanings of two words. Based on this metric, communication of a finite set of words in the context of Natural Language Processing (NLP) is modeled as a Bayesian game from game theory and optimized for improved semantic transmission over a binary symmetric channel.

Recently, enabled by the rise of ML in communications research, deep learning-based NLP techniques, i.e., transformer networks, were introduced in [17], [18], [19], [20] for the task of text and speech transmission. Although recovery of the messages is the main objective, the aim of these techniques is to learn compressed hidden representations of the semantic content of sentences. This leads to performance improvements in semantic metrics, especially at low SNR compared to classical digital transmissions.

For the first time, the authors in [21] showed that task-oriented communications (Level C) for edge cloud transmission can be mathematically formulated as an Information Bottleneck (IB) optimization problem. Moreover, for solving the IB problem, they introduce a Deep Neural Network (DNN) - based approach and show its applicability for task-oriented communications, i.e., for the specific task of edge cloud transmission. But they do not consider semantic communications in particular. The terminus *"semantic information"* is only mentioned once in [21] referring to Joint Source Channel Coding (JSCC) of text from [17] using recurrent neural networks. In [17], the authors observe that sentences that express the same idea have embeddings that are close together in Hamming distance. But they use cross entropy between words and estimated words as the loss function and use the word error rate as the performance measure, which both do not reflect if two sentences have the same meaning but rather that both are exactly the same.

As a result, semantic communication is still a nascent field: It remains still unclear what this term exactly means and especially its distinction from Joint Source Channel Coding (JSCC) [17], [22]. As a result, many survey papers aim to provide an interpretation, see, e.g., [7], [8], [9], [10]. We will revisit this issue in Sec. II.

B. Main Contributions

The main contributions of this article are:

- Motivated by the approach of Bao, Basu et al. [13], [14], we adopt the terminus of a semantic source. Inspired by Weaver's notion, we bring it to the context of communications by considering the complete Markov chain including semantic source, communications source, transmit signal, wireless channel, and received signal in contrast to both [13], [14]. Further, we also extend beyond the example of deterministic entailment relations between "models" and "messages" based on propositional logic in [13], [14] to probabilistic semantic channels.
- We define the task of semantic communication in the sense that we perform data compression, coding, and transmission of messages observed such that the semantic Random Variable (RV) at a recipient is best preserved. Basically, we implement joint source-channel coding of messages conveying the semantic RV, but not differentiating between Levels A and B. We formulate the semantic communication design either as an Information Maximization or as an Information Bottleneck (IB) optimization problem [23], [24], [25].
  - Although the approach pursued here again leads to an IB problem as in [21], our article introduces a new classification and perspective of semantic communication and different ML-based solution approaches. Different from [21], we derive an exact solution to the IB problem that maximizes the mutual information for a fixed encoder output dimension that bounds the compression rate.
  - The publication presented here differs also both in the interpretation of what is meant by semantic information and in the objective of recovering this semantic information from approaches to semantic communication presented in the literature like, e.g., [19].
- Finally, we propose the ML-based semantic communication system SINFONY for a distributed multipoint scenario: SINFONY communicates the meaning behind multiple messages that are observed at different senders to a single receiver for semantic recovery.
- We analyze SINFONY by processing images as an example of messages. Notably, numerical results reveal a tremendous rate-normalized SNR shift up to 20 dB compared to classically designed communication systems.

In the following, we reinterpret Weaver's philosophical considerations in Sec. II-A paving the way for our proposed theoretical framework in Sec. II. Finally, in Sec. III and IV, we provide one numerical example of semantic communication, i.e., SINFONY, and summarize the main results, respectively.

II. A Framework for Semantics

A. Philosophical Considerations

Despite the much-renewed interest, research on semantic communication is still in its infancy and recent work reveals a differing understanding of the word *semantics*. In this work, we contribute our interpretation. To motivate it, we shortly revisit the research birth hour of communications from a philosophical point of view: Its theoretical foundation was laid by Shannon in his landmark paper [1] in 1948.

He stated that *"Frequently the messages have meaning; that is they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem."* In fact, this viewpoint abstracts all kinds of information one may transmit, e.g., oral and written speech, sensor data, etc., and lays also the foundation for the research area of Shannon information theory. Thus, it found its way into many other research areas where data or information is processed, including Artificial Intelligence (AI) and especially its subdomain Machine Learning (ML).

Weaver saw this broad applicability of Shannon's theory back in 1949. In his comprehensible review of [1], he first states that *"there **seem** to be [communication] problems at three levels"* [2] already mentioned in Sec. I-A. These three levels are quoted in recent works, where Level C is oftentimes referred to as goal-oriented communication instead [8].

But we note that in his concluding section, he then questions this segmentation: He argues for the generality of the theory at Level A for all levels and ***"that the interrelation of the three levels is so considerable that one's final conclusion may be that the separation into the three levels is really artificial and undesirable"***.

It is important to emphasize that the separation is rather arbitrary. We agree with Weaver's statement because the most important point that is also the focus herein is the definition of the term semantics, e.g., by Basu et al. [13], [14]. Note that the entropy of the semantics is less than or equal to the entropy of the messages. Consequently, we can save information rate by introducing meaning or context. In fact, we are able to add arbitrarily many levels of semantic details to the communication problem and optimize communications for a specific semantic background, e.g., an application or human.

B. Semantic System Model

1) Semantic Source and Channel: Now, we will define our information-theoretic system model of semantic communication. Fig. 1 shows the schematic of our model. We assume the existence of a semantic source, described as a hidden target multivariate Random Variable (RV) $\mathbf{z} \in \mathcal{M}_z^{N_z \times 1}$ from a domain $\mathcal{M}_z$ of dimension N_z distributed according to a probability density or mass function (pdf/pmf) $p(\mathbf{z})$. To simplify the discussion, we assume it to be discrete and memoryless.[1]

This approach is similar to that of [13], [14]: In [13], [14], the semantic source[2] is described by "models of the world". In [14], a semantic channel then generates messages through entailment relations between "models" and "messages". We will call these "messages" source signal and define it to be a RV $\mathbf{s} \in \mathcal{M}_s^{N_s \times 1}$ as it is usually observed and enters the communication system. In classic Shannon design, the aim is to reconstruct the source $\mathbf{s}$ as accurately as possible at the receiver side. Further, we note that the authors in [14] considered the example of a semantic channel with deterministic entailment relations between $\mathbf{z}$ and $\mathbf{s}$ based on propositional logic. In this article, we go beyond this assumption and consider probabilistic semantic channels modeled by distribution $p(\mathbf{s}|\mathbf{z})$ that include the entailment in [14] as special cases, i.e., $p(\mathbf{s}|\mathbf{z}) = \delta(\mathbf{s} - f(\mathbf{z}))$ where $\delta(\cdot)$ is the Dirac delta function and $f(\cdot)$ is any generic function. Our viewpoint is motivated by the recent success of pattern recognition tools that advanced the field of AI in the 2010s and may be used to extract semantics [5].

Our approach also extends models as in [19]. There, the authors design a semantic communication system for the transmission of written language/text similar to [17] using transformer networks. In contrast to our work, [19] does not define meaning as RV $\mathbf{z}$. The objective in [19] is to reconstruct $\mathbf{s}$ (sentences) as well as possible, rather than the meaning (RV $\mathbf{z}$) conveyed in $\mathbf{s}$. Optimization is done w.r.t. to a loss function consisting of two parts: Cross entropy between language input $\mathbf{s}$ and output $\hat{\mathbf{s}}$, as well as a scaled mutual information term between transmit signal $\mathbf{x}$ and receive signal $\mathbf{y}$. After optimization, the authors measure semantic performance with BLEU and semantic similarity.

We now provide an example to explain what we understand under a semantic source $\mathbf{z}$ and channel $p(\mathbf{s}|\mathbf{z})$: Let us assume a biologist who has an image of a tree. The biologist wants to know what kind of tree it is by interpreting the observed data (image). In this case, the semantic source $\mathbf{z}$ is a multivariate RV composed of a categorical RV with M tree classes. For any realization (sample value) $\mathbf{z}_i$ of the semantic source, the semantic channel $p(\mathbf{s}|\mathbf{z})$ then outputs with some probability one image $\mathbf{s}_i$ of a tree conveying characteristics of $\mathbf{z}$, i.e., its meaning. Note that the underlying meaning of the same sensed data (message) can be different for other recipients, e.g., humans or tasks/applications, i.e., in other semantic contexts. Imagine a child, i.e., a person with different characteristics (personality, expertise, knowledge, goals, and intentions) than the biologist, who is only interested if he/she can climb up this tree or whether the tree provides shade. Thus, we include the characteristics of the sender and receiver in RV $\mathbf{z}$ and consider it directly in compression and encoding.

Compared to [13], we, therefore, argue that we also include level C by semantic source and channel since context can be included on increasing layers of complexity. First, a RV $\mathbf{z}_1$ might capture the interpretation, like the classification of images or sensor data. Moving beyond the first semantic layer, then a RV $\mathbf{z}_2$ might expand this towards a more general goal, like keeping a constant temperature in power plant control. In fact, we can add or remove context, i.e., semantics and goals, arbitrarily often according to the human or application behind, and we can optimize the overall (communication) system w.r.t. $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_i$, respectively.

As a last remark, we note that we basically defined probabilistic semantic relationships, and it remains the question of how exactly they might look. In our example, the meaning of the images needs to be labeled into real-world data pairs $\{\mathbf{s}_i, \mathbf{z}_i\}$ by experts/humans, since image recognition lacks precise mathematical models. This is also true for NLP [19]:

[1] For the remainder of the article, note that the domain of all RVs $\mathcal{M}$ may be either discrete or continuous. Further, we note that the definition of entropy for discrete and continuous RVs differs. For example, the differential entropy of continuous RVs may be negative whereas the entropy of discrete RVs is always positive [26]. Without loss of generality, we will thus assume all RVs either to be discrete or to be continuous. In this work, we avoid notational clutter by using the expected value operator: Replacing the integral by summation over discrete RVs, the equations are also valid for discrete RVs and vice versa.

[2] Note that in [14] the semantic information source is defined as a tuple $(\mathbf{z}, \mathbf{s}, p(\mathbf{z}, \mathbf{s}), L)$. In this original notation, $\mathbf{z}$ is the model, $\mathbf{s}$ the message, $p(\mathbf{z}, \mathbf{s})$ the joint distribution of $\mathbf{z}$ and $\mathbf{s}$, and L is the deterministic formal language.

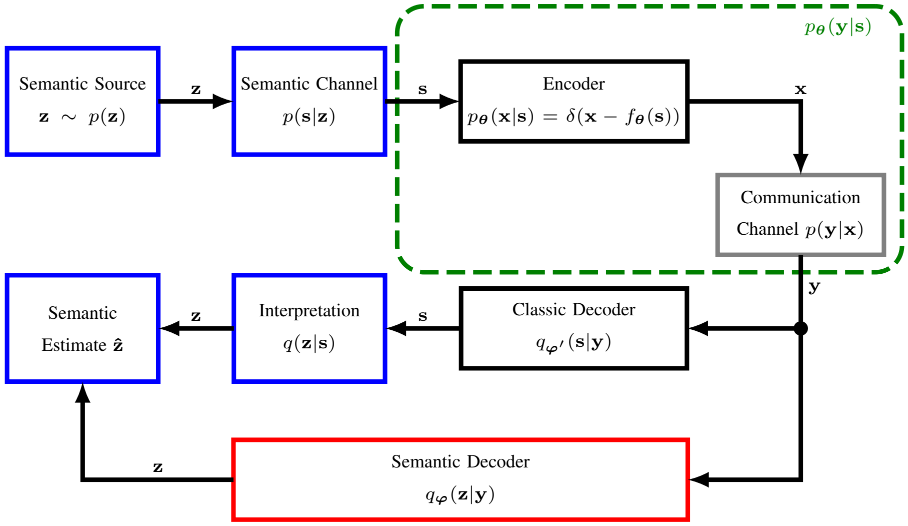

Fig. 1. Block diagram of the considered semantic system model.

How can we measure if two sentences have the same meaning, i.e., how does the semantic space look like? In contrast, in [14], the authors are able to solve their well-defined technical problem (motion detection) by a model-driven approach. We can thus distinguish between model and data-driven semantics, which both can be handled within Shannon's information theory.

2) Semantic Channel Encoding: After the semantic source and channel in Fig. 1, we extend upon [13] by differentiating between "message"/source signal $\mathbf{s}$ and transmit signal $\mathbf{x} \in \mathcal{M}_x^{N_{\text{Tx}}\times 1}$. Our challenge is to encode the source signal $\mathbf{s}$ onto the transmit signal vector $\mathbf{x}$ for reliable semantic communication through the physical communication channel $p(\mathbf{y}|\mathbf{x})$, where $\mathbf{y} \in \mathcal{M}_y^{N_{\text{Rx}}\times 1}$ is the received signal vector. We assume the encoder $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$ to be parametrized by a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^{N_{\boldsymbol{\theta}}\times 1}$. Note that $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$ is probabilistic here, but assumed to be deterministic in communications with $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s}) = \delta(\mathbf{x} - \mu_{\boldsymbol{\theta}}(\mathbf{s}))$ and encoder function $\mu_{\boldsymbol{\theta}}(\mathbf{s})$.

In summary, in contrast to both [13], [14], we consider the complete Markov chain $\mathbf{z} \leftrightarrow \mathbf{s} \leftrightarrow \mathbf{x} \leftrightarrow \mathbf{y}$ including semantic source $\mathbf{z}$, communications source $\mathbf{s}$, transmit signal $\mathbf{x}$ and receive signal $\mathbf{y}$. By this means, we distinguish from [14] which only deals with semantic compression, and [13] which is about joint semantic compression and channel coding (Level B). In [13], the authors consider the classic transmission system (Level A) as the (semantic) channel (not to be confused with the definition of the semantic channel in [14] which we make use of in this publication).

At the receiver side, one approach is maximum a-posteriori decoding w.r.t. RV $\mathbf{s}$ that uses the posterior $p_{\boldsymbol{\theta}}(\mathbf{s}|\mathbf{y})$, being deduced from prior $p(\mathbf{s})$ and likelihood $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})$ by application of Bayes law. Based on the estimate of $\mathbf{s}$, then the receiver interprets the actual semantic content $\mathbf{z}$ by $p(\mathbf{z}|\mathbf{s})$.

Another approach we propose is to include the semantic hidden target RV $\mathbf{z}$ into the design by processing $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})$. If the calculation of the posterior is intractable, we can replace $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})$ by the approximation $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$, i.e., the semantic decoder, with parameters $\boldsymbol{\varphi} \in \mathbb{R}^{N_{\boldsymbol{\varphi}}\times 1}$. We expect the following benefit: We assume the entropy $\mathcal{H}(\mathbf{z}) = \mathrm{E}_{\mathbf{z}\sim p(\mathbf{z})}[-\ln p(\mathbf{z})]$[3] of the semantic RV $\mathbf{z}$, i.e., the actual semantic uncertainty or information content, to be less or equal to the entropy $\mathcal{H}(\mathbf{s})$ of the source $\mathbf{s}$, i.e., $\mathcal{H}(\mathbf{z}) \leq \mathcal{H}(\mathbf{s})$. Consequently, since we would like to preserve the relevant, i.e., semantic, RV $\mathbf{z}$ rather than $\mathbf{s}$, we can compress more s.t. preserving $\mathbf{z}$ conveyed in $\mathbf{s}$. Note that in semantic communication the relevant variable is $\mathbf{z}$, not $\mathbf{s}$. Thus, processing $p_{\boldsymbol{\theta}}(\mathbf{s}|\mathbf{y})$ without taking $\mathbf{z}$ into consideration resembles the classical approach. Instead of using (and transmitting) $\mathbf{s}$ for inference of $\mathbf{z}$, we now want to find a compressed representation $\mathbf{y}$ of $\mathbf{s}$ containing the relevant information about $\mathbf{z}$.

C. Semantic Communication Design via Infomax Principle

After explaining the system model and the basic components, we are able to approach a semantic communication system design: We first define an optimization problem to obtain the encoder $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$ following the Infomax principle from an information theoretic perspective [26]. Thus, we like to find the distribution $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$ that maps $\mathbf{s}$ to a representation $\mathbf{x}$ such that at most information of the relevant RV $\mathbf{z}$ is included in $\mathbf{y}$, i.e., we maximize the Mutual Information (MI) $I(\mathbf{z};\mathbf{y})$ w.r.t. $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$ [27]:

$$\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\ I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y}) \tag{1}$$

$$=\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\ \mathrm{E}_{\mathbf{z},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}\left[\ln \frac{p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}{p(\mathbf{z})p_{\boldsymbol{\theta}}(\mathbf{y})}\right] \tag{2}$$

$$=\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\ \mathcal{H}(\mathbf{z}) - \mathcal{H}(p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y}), p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})) \tag{3}$$

$$=\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\ \mathrm{E}_{\mathbf{z},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}[\ln p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})]\ . \tag{4}$$

There, $\mathcal{H}(p(\mathbf{x}), q(\mathbf{x})) = \mathrm{E}_{\mathbf{x}\sim p(\mathbf{x})}[-\ln q(\mathbf{x})]$ is the cross entropy between two pdfs/pmfs $p(\mathbf{x})$ and $q(\mathbf{x})$. Note independence from $\boldsymbol{\theta}$ in $\mathcal{H}(\mathbf{z})$ and dependence in $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})$ and

[3]There, $\mathrm{E}_{\mathbf{x}\sim p(\mathbf{x})}[f(\mathbf{x})]$ denotes the expected value of $f(\mathbf{x})$ w.r.t. both discrete or continuous RVs $\mathbf{x}$.

$p_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{y})$ through the Markov chain $\mathbf{z} \rightarrow \mathbf{s} \rightarrow \mathbf{y}$. It is worth mentioning that we so far have not set any constraint on the variables we deal with. Hence, the form of $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})$ has to be constrained to avoid learning a trivial identity mapping $\mathbf{y} = \mathbf{s}$. We indeed constrain the optimization by our communication channel $p(\mathbf{y}|\mathbf{x})$ we assume to be given.

If the calculation of the posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})$ in (4) is intractable, we are able to replace it by a variational distribution $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$ with parameters $\boldsymbol{\varphi}$. Similar to the transmitter, Deep Neural Networks (DNNs) are usually proposed [19], [28] for the design of the approximate posterior $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$ at the receiver. To improve the performance complexity trade-off, the application of *deep unfolding* can be considered, a model-driven learning approach that introduces model knowledge of $p_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{x}, \mathbf{y}, \mathbf{z})$ to create $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$ [6], [29]. With $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$, we are able to define a Mutual Information Lower BOund (MILBO) [27] similar to the well-known Evidence Lower BOund (ELBO) [5]:

$$\begin{aligned} I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y}) &\geq \mathrm{E}_{\mathbf{z},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}[\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})] && (5)\\ &= \mathrm{E}_{\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y})}\left[\mathrm{E}_{\mathbf{z}\sim p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})}[\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})]\right] && (6)\\ &= -\,\mathrm{E}_{\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y})}[\mathcal{H}\left(p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y}), q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right)] && (7)\\ &= -\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}\,. && (8) \end{aligned}$$

The lower bound holds since $-\mathcal{H}\left(p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y}), p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})\right)$ itself is a lower bound of the expression in (3) and $\mathrm{E}_{\mathbf{z},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}[\ln p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})/\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})] \geq 0$.

Thus, the idea is to learn parametrizations of the transmitter discriminative model and of the variational receiver posterior, e.g., by Auto Encoders (AEs) or reinforcement learning. Note that in our semantic problem (1), we do not auto encode the hidden $\mathbf{z}$ itself, but encode $\mathbf{s}$ to obtain $\mathbf{z}$ by decoding. This can be seen from Fig. 1 and by rewriting the amortized cross entropy (7), (8):

$$\begin{aligned} \mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}} &= \mathrm{E}_{\mathbf{y}\sim p(\mathbf{y})}[\mathcal{H}\left(p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y}), q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right)] && (9)\\ &= \mathrm{E}_{\mathbf{s},\mathbf{x},\mathbf{y},\mathbf{z}\sim p_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{x},\mathbf{y},\mathbf{z})}[-\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})] && (10)\\ &= \mathrm{E}_{\mathbf{s},\mathbf{z}\sim p(\mathbf{s},\mathbf{z})}\left[\mathrm{E}_{\mathbf{x}\sim p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}\left[\mathrm{E}_{\mathbf{y}\sim p(\mathbf{y}|\mathbf{x})}[-\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})]\right]\right]\,. \end{aligned}$$

We can further prove the amortized cross entropy to be decomposable into

$$\begin{aligned} &\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}\\ &= \mathrm{E}_{\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y})}\left[\mathrm{E}_{\mathbf{z}\sim p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})}[-\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y}) + \ln p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})\right.\\ &\qquad \left.-\ln p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})]\right] && (11)\\ &= \mathrm{E}_{\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y})}[D_{\mathrm{KL}}\left(p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y}) \parallel q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right)] + \underbrace{\mathcal{H}(\mathbf{z}|\mathbf{y})}_{=-I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})+\mathcal{H}(\mathbf{z})} && (12)\\ &= \mathcal{H}(\mathbf{z}) - \underbrace{I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})}_{\text{enc. objective}} + \underbrace{\mathrm{E}_{\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y})}[D_{\mathrm{KL}}\left(p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y}) \parallel q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right)]}_{\text{dec. objective}}\,. && (13) \end{aligned}$$

To the end, maximization of the MILBO w.r.t. $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$ balances maximization of the mutual information $I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})$ and minimization of the Kullback-Leibler (KL) divergence $D_{\mathrm{KL}}\left(p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y}) \parallel q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right)$. The former objective can be seen as a regularization term that favors encoders with high mutual information, for which decoders can be learned that are close to the true posterior.

D. Classical Design Approach

If we consider classical communication design approaches, we would solve the problem

$$\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\; I\left(\mathbf{s};\mathbf{y}\right) \tag{14}$$

which relates to Joint Source Channel Coding (JSCC). There, the aim is to find a representation $\mathbf{x}$ that retains a significant amount of information about the source signal $\mathbf{s}$ in $\mathbf{y}$. Again, we can apply the lower bound (8). In fact, bounding Eq. (14) by (8) shows that approximate maximization of the mutual information justifies the minimization of the cross entropy in the Auto Encoder (AE) approach [4] oftentimes seen in recent wireless communication literature [4], [22], [17].

E. Information Bottleneck View

It should be stressed that we have not set any constraints on the variables in the Infomax problem so far. However, in many applications, compression is needed because of the limited information rate. Therefore, we can formulate an optimization problem where we like to maximize the relevant information $I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})$ subject to the constraint to limit the compression rate $I_{\boldsymbol{\theta}}(\mathbf{s};\mathbf{x})$ to a maximum information rate I_{C}:

$$\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\; I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y}) \quad \text{s.t.} \quad I_{\boldsymbol{\theta}}(\mathbf{s};\mathbf{x}) \leq I_{\mathrm{C}}\,. \tag{15}$$

Problem (15) is called the Information Bottleneck (IB) problem [8], [23], [30], [31]. The IB method introduced by Tishby et al. [23] has been the subject of intensive research for years and has proven to be a suitable mathematical/information-theoretical framework for solving numerous problems - also in wireless communications [32], [33], [24], [25]. Note that we aim for an encoder that compresses $\mathbf{s}$ into a compact representation $\mathbf{x}$ for discrete RVs by clustering and for continuous RVs by dimensionality reduction.

To solve the constrained optimization problem (15), we can use Lagrangian optimization and obtain

$$\underset{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})}{\arg\max}\; I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y}) - \beta I_{\boldsymbol{\theta}}(\mathbf{s};\mathbf{x}) \tag{16}$$

with Lagrange multiplier $\beta \geq 0$. The Lagrange multiplier β allows defining a trade-off between the relevant information $I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})$ and compression rate $I_{\boldsymbol{\theta}}(\mathbf{s};\mathbf{x})$, which indicates the relation to rate distortion theory [24]. With $\beta = 0$, we have the InfoMax problem (1) whereas for $\beta \rightarrow \infty$ we minimize compression rate. Calculation of the mutual information terms may be computationally intractable, as in the InfoMax problem (1). Approximation approaches can be found in [34], [35]. Notable exceptions include if the RVs are all discrete or Gaussian distributed.

We note that in [8], [21] the authors already introduced the IB problem to task-oriented communications. But [8], [21] do not address our viewpoint or classification: We compress and channel encode the messages/communications source $\mathbf{s}$ for given entailment $p(\mathbf{s}|\mathbf{z})$, in the sense of a data-reduced and reliable communication of the semantic RV $\mathbf{z}$. Basically, we implement joint source-channel coding of $\mathbf{s}$ s.t. preserve the semantic RV $\mathbf{z}$ and we do not differentiate between Levels A and B as indicated by Weaver's notion outlined in I-A. Indeed, we draw a direct connection to IB compared to related semantic communication literature [17], [19], [28] that so far only included optimization with terms reminiscent of the IB problem.

This article does not only distinct itself on a conceptual, but also on a technical level from [21], [36]: We follow a different strategy to solve (15). First, let us assume the RVs to be discrete. Indeed, this is true if the RVs are processed discretely with finite resolution on digital signal processors, as in the numerical example of Sec. III. Then, we can upper bound compression rate $I\left(\mathbf{s};\mathbf{x}\right)$ by the sum of entropies of any output x_n of the encoder $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{s})$:

$$I_{\boldsymbol{\theta}}\left(\mathbf{s};\mathbf{x}\right)=\mathcal{H}\left(\mathbf{x}\right)-\underbrace{\mathcal{H}\left(\mathbf{x}|\mathbf{s}\right)}_{\geq 0}\leq\mathcal{H}\left(\mathbf{x}\right)\leq\sum_{n=1}^{N_{\mathrm{Tx}}}\mathcal{H}\left(x_n\right)=I_{\mathrm{C}}\,. \tag{17}$$

Note that the entropy sum in (17) grows with N_{Tx}. By choosing the encoder output dimension N_{Tx}, we can thus set the constraint I_{C} higher or lower.

To the end, we set constraint I_{C} by fixing N_{Tx}. With fixed constraint I_{C}, we maximize the relevant information $I_{\boldsymbol{\theta}}\left(\mathbf{z};\mathbf{y}\right)$. By doing so, we derive an exact solution that maximizes $I_{\boldsymbol{\theta}}\left(\mathbf{z};\mathbf{y}\right)$ for a fixed encoder output dimension that bounds the compression rate. As in the InfoMax problem, we can exploit the MILBO to use the amortized cross entropy $\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}$ in (9) as the optimization criterion. In [21], however, a variational approximation of (16) is solved by approximating the KL divergence term of the variational IB problem. Further, the authors use a log-uniform distribution as the variational prior in [21] to induce sparsity so that the number of outputs is dynamically determined based on the channel condition, i.e., SNR.

F. Implementation Considerations

Now, we will provide important implementation considerations for optimization of (8)/(10) and (15). We note that computation of the MILBO leads to similar problems as for the ELBO [26]. If calculating the expected value in (10) cannot be solved analytically or is computationally intractable, we can approximate it using Monte Carlo sampling techniques. Then, our loss function amounts to

$$\begin{aligned}\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}&=\mathrm{E}_{\mathbf{z},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{z},\mathbf{y})}\left[-\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right]\\&\approx-\frac{1}{N}\sum_{i=1}^{N}\ln q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y}_i)\,.\end{aligned} \tag{18}$$

For Stochastic Gradient Descent (SGD) - based optimization like, e.g., in the AE approach, the gradient w.r.t. $\boldsymbol{\varphi}$ can then be calculated by

$$\frac{\partial}{\partial\boldsymbol{\varphi}}\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}=\frac{\partial}{\partial\boldsymbol{\varphi}}\mathrm{E}_{\mathbf{z},\mathbf{s},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})p(\mathbf{s}|\mathbf{z})p(\mathbf{z})}\left[-\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right] \tag{19}$$

$$\approx-\frac{1}{N}\sum_{i=1}^{N}\frac{\partial\ln q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y}_i)}{\partial\boldsymbol{\varphi}} \tag{20}$$

and by application of the backpropagation algorithm to $\frac{\partial}{\partial\boldsymbol{\varphi}}\ln q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y}_i)=\frac{\partial}{\partial\boldsymbol{\varphi}}q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y}_i)/q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y}_i)$ in Automatic Differentiation Frameworks (ADF), e.g., TensorFlow and PyTorch. Computation of the so-called Reinforce gradient w.r.t. $\boldsymbol{\theta}$ leads to a high variance of the gradient estimate since we sample w.r.t. the distribution $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})$ dependent on $\boldsymbol{\theta}$ [26].

Typically, the reparametrization trick is used to overcome this problem, as in the VAE approach [26]. Here it is applicable if the latent variable $\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})$ can be decomposed into a differentiable function $f_{\boldsymbol{\theta}}(\mathbf{n},\mathbf{s})$ and a RV $\mathbf{n}\sim p(\mathbf{n})$ independent of $\boldsymbol{\theta}$. Fortunately, the typical forward model of a communication system $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})$ fulfills this criterion. Assuming a deterministic DNN encoder $\mu_{\boldsymbol{\theta}}(\mathbf{s})$ and additive noise $\mathbf{n}$ with covariance $\boldsymbol{\Sigma}$, we can thus rewrite into $f_{\boldsymbol{\theta}}(\mathbf{n},\mathbf{s})=\mu_{\boldsymbol{\theta}}(\mathbf{s})+\boldsymbol{\Sigma}^{1/2}\cdot\mathbf{n}$ and accordingly the Monte Carlo approximation of the amortized cross entropy gradient into:

$$\begin{aligned}&\frac{\partial}{\partial\boldsymbol{\theta}}\mathcal{L}^{\mathrm{CE}}_{\boldsymbol{\theta},\boldsymbol{\varphi}}\\&=-\frac{\partial}{\partial\boldsymbol{\theta}}\mathrm{E}_{\mathbf{z},\mathbf{s},\mathbf{y}\sim p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{s})p(\mathbf{s}|\mathbf{z})p(\mathbf{z})}\left[\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})\right] &(21)\\&=-\mathrm{E}_{\mathbf{z},\mathbf{s},\mathbf{n}\sim p(\mathbf{n})p(\mathbf{s}|\mathbf{z})p(\mathbf{z})}\left[\frac{\partial f_{\boldsymbol{\theta}}(\mathbf{n},\mathbf{s})}{\partial\boldsymbol{\theta}}\cdot\frac{\partial\ln q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})}{\partial\mathbf{y}}\right] &(22)\\&\approx-\frac{1}{N}\sum_{i=1}^{N}\frac{\partial f_{\boldsymbol{\theta}}(\mathbf{n}_i,\mathbf{s}_i)}{\partial\boldsymbol{\theta}}\cdot\left.\frac{\partial\ln q_{\boldsymbol{\varphi}}(\mathbf{z}_i|\mathbf{y})}{\partial\mathbf{y}}\right|_{\mathbf{y}=f_{\boldsymbol{\theta}}(\mathbf{n}_i,\mathbf{s}_i)}\,. &(23)\end{aligned}$$

The trick is easy to implement in ADFs by adding a simple noise layer after (DNN) function $\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{s})$. The noise layer is usually used for regularization in ML literature. Thus, in recent works, e.g., [4], unsupervised optimization of AEs is treated like a supervised learning problem.

III. Example of Semantic Information Recovery

In this section, we provide one numerical example of data-driven semantics to explain what we understand under a semantic communication design and to show its benefits: It is the task of image classification. In fact, we consider our

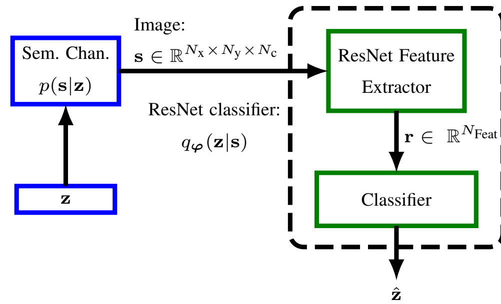

Fig. 2. Central image processing: Based on the images, ResNet extracts semantics by classification.

example of the biologist from Sec. II-B who wants to know which type the tree is.

For the remainder of this article, we will thus assume the hidden semantic RV to be a one-hot vector $\mathbf{z} \in \{0,1\}^{M\times 1}$ where all elements are zero except for one element representing one of the M image classes. Then, the semantic channel $p(\mathbf{s}|\mathbf{z})$ (see Fig. 1) generates images belonging to this class, i.e., the source signal $\mathbf{s}$.

Note that for point-to-point transmission we could first classify the image based on the posterior $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{s})$ as shown in Fig. 2 and transmit the estimate $\hat{\mathbf{z}}$ (encoded into $\mathbf{x}$) through the physical channel since this would be most rate or bandwidth efficient.

But if the image information is distributed across multiple agents, all (sub) images may contribute useful information for classification. We could thus lose information when making hard decisions on each transmitter's side. In the distributed setting, transmission and combination of features, i.e., soft information, is crucial to obtain high classification accuracy.

Further, we note that transmission of full information, i.e., raw image data $\mathbf{s}$, through a wireless channel from each agent to a central unit for full image classification would consume a lot of bandwidth. This case is also shown in Fig. 2 assuming perfect communication links between the output of the semantic channel and the input of the ResNet Feature Extractor.

Therefore, we investigate a distributed setting shown in Fig. 3. There, each of four agents sees its own image $\mathbf{s}_1, \dots, \mathbf{s}_4 \sim p(\mathbf{s}_i|\mathbf{z})$ being generated by the same semantic RV $\mathbf{z}$. Based on these images, a central unit shall extract semantics, i.e., perform classification. We propose to optimize the four encoders $p_{\boldsymbol{\theta}_i}(\mathbf{x}_i|\mathbf{s}_i)$ with $i = 1, \dots, 4$, each consisting of a bandwidth efficient feature extractor (ResNet Feature Extractor i) and transmitter (Tx i) **jointly** with a decoder $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$, consisting of a receiver (Rx) and concluding classifier (Classifier), w.r.t. cross entropy (10) of the semantic labels (see Fig. 3). In the end, we maximize the system's overall semantic measure, i.e., classification accuracy. Note that this scenario is different from [37]: We include a physical communication channel (Comm. Channel i) since we aim to transmit and not only compress. For the sake of simplicity, we assume orthogonal channel access. The IB is addressed by limiting the number of channel uses, which defines the constraint I_{C} in (15).

As a first demonstration example, we use the grayscale MNIST and colored CIFAR10 datasets with $M = 10$ image classes [38]. We assume that the semantic channel generates an image that we divide into four equally sized quadrants and each agent observes one quadrant $\mathbf{s}_1, \dots, \mathbf{s}_4 \in \mathbb{R}^{N_{\mathrm{x}}\times N_{\mathrm{y}}\times N_{\mathrm{c}}}$ where N_{x} and N_{y} is the number of image pixels in the x- and y-dimension, respectively, and N_{c} is the number of color channels. Albeit this does not resemble a realistic scenario, note that we can still show the basic working principle and ease implementation.

A. ResNet

For the design of the overall system, we rely on a famous DNN approach for feature extraction, breaking records at the time of invention: ResNet [38], [39]. The key idea of ResNet is that it consists of multiple residual units: Each unit's input is fed directly to its output and if the dimensions do not match, a convolutional layer is used. This structure allows for fast training and convergence of DNNs since the training error can be backpropagated to early layers through these skip connections. From a mathematical point of view, usual DNNs have the design flaw that using a larger function class, i.e., more DNN layers, does not necessarily increase the expressive power. However, this holds for nested functions like ResNet which contain the smaller classes of early layers.

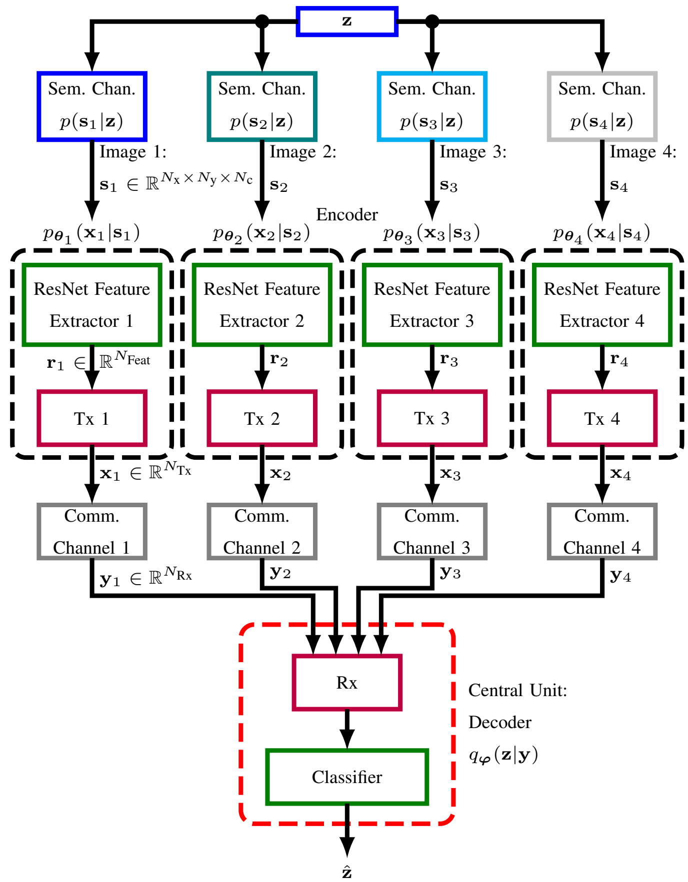

Fig. 3. Semantic INFOrmation traNsmission and recoverY (SINFONY) for distributed agents. Each agent extracts features for bandwidth-efficient transmission. Based on the received signal, the central unit extracts semantics by classification.

Each residual unit itself consists of two Convolutional NNs (CNNs) with subsequent batch normalization and ReLU activation function, i.e., $\rho_1(\cdot) = \max(\cdot, 0)$, to extract translation invariant and local features across two spatial dimensions N_{x} and N_{y}. Color channels like in CIFAR10 add a third dimension $N_{\mathrm{c}} = 3$ and additional information. The idea behind stacking multiple layers of CNNs is that features tend to become more abstract from early layers (e.g., edges and circles) to final layers (e.g., beaks or tires).

In this work, we use the preactivation version of ResNet without bottlenecks from [38], [39] implemented for classification on the dataset CIFAR10. In Tab. I, we show its structure for the distributed scenario from Fig. 3. There, ResNetBlock is

TABLE I
SEMANTIC INFORMATION TRANSMISSION AND RECOVERY (SINFONY) - DNN ARCHITECTURE FOR DISTRIBUTED IMAGE CLASSIFICATION.

Component	Layer	Dimension
Input	Image (MNIST, CIFAR10)	$(14, 14, 1)$, $(16, 16, 3)$
4×	Conv2D	$(14, 14, 14)$, $(16, 16, 16)$
Feature	ResNetBlock (2/3 res. un.)	$(14, 14, 14)$, $(16, 16, 16)$
Extractor	ResNetBlock (2/3 res. un.)	$(7, 7, 28)$, $(8, 8, 32)$
	ResNetBlock (2/3 res. un.)	$(4, 4, 56)$, $(4, 4, 64)$
	Batch Normalization	$(4, 4, 56)$, $(4, 4, 64)$
	ReLU activation	$(4, 4, 56)$, $(4, 4, 64)$
	GlobalAvgPool2D	(56), (64)
4× Tx	ReLU	N_{Tx}
	Linear	N_{Tx}
	Normalization (dim.)	N_{Tx}
Channel	AWGN	N_{Tx}
Rx	ReLU (4× shared)	$(2, 2, N_{\text{w}})$
	GlobalAvgPool2D	N_{w}
Classifier	Softmax	10

the basic building block of the ResNet architecture. Each block consists of multiple residual units (res. un.) and we use 2 for MNIST and 3 for CIFAR10, which means we use ResNet14 and ResNet20, respectively. We arrive at the architecture of central image processing from Fig. 2 by removing the components Tx, (physical) Channel, and Rx and increasing each spatial dimension by 2 to contain all quadrants of the original image. For further implementation details, we refer the reader to the original work [39].

B. Distributed Semantic Communication Design Approach

Our key idea here is to modify ResNet w.r.t. the communication task by splitting it at a suitable point where a representation $\mathbf{r} \in \mathbb{R}^{N_{\text{Feat}} \times 1}$ of semantic information with low-bandwidth is present (see Fig. 2 and 3). ResNet and CNNs in general can be interpreted to extract features: With full images, we obtain a feature map of size $8 \times 8 \times N_{\text{Feat}}$ after the last ReLU activation (see Tab. I). These local features are aggregated by the global average pooling layers across the 2 spatial dimensions into $\mathbf{r}$. Based on these N_{Feat} global features in $\mathbf{r}$, the softmax layer finally classifies the image. We note that the features contain the relevant information w.r.t. the semantic RV $\mathbf{z}$ and are of low dimension compared to the original image or even its sub-images, i.e., 64 compared to $16 \times 16 \times 3 = 768$ for CIFAR10.

Therefore, we aim to transmit each agent's local features $\mathbf{r}_i \in \mathbb{R}^{N_{\text{Feat}} \times 1}$ $(i = 1, \dots, 4)$ instead of all sub-images $\mathbf{s}_i$ and add the component Tx in Tab. I to encode the features $\mathbf{r}_i \in \mathbb{R}^{N_{\text{Feat}} \times 1}$ into $\mathbf{x}_i \in \mathbb{R}^{N_{\text{Tx}} \times 1}$ for transmission through the wireless channel (see Fig. 3). We note that $\mathbf{x}_i \in \mathbb{R}^{N_{\text{Tx}} \times 1}$ is analog and that the output dimension N_{Tx} of $\mathbf{x}_i$ defines the number of channel uses per agent/image. Note that the less often we use the wireless channel (N_{Tx}), the less information we transmit but the less bandwidth we consume, and vice versa. Hence, the number of channel uses defines the IB in (15). We implement the Tx module by DNN layers. To limit the transmit power to one, we constrain the Tx output by the norm along the training batch or the encode vector dimension (dim.), e.g., $\mathbf{x}_i = \sqrt{N_{\text{Tx}}} \cdot \tilde{\mathbf{x}}_i / \|\tilde{\mathbf{x}}_i\|_2$ where $\tilde{\mathbf{x}}_i \in \mathbb{R}^{N_{\text{Tx}} \times 1}$ is the output of the layer Linear from Tab. I. For numerical simulations, we choose all Tx layers to have width N_{Tx}.

At the receiver side, we use a single Rx module only with shared DNN layers and parameters φ for all inputs $\mathbf{y}_i$: This setting would be optimal if any feature is reflected in any sub-image and if the statistics of the physical channels are the same. Exploiting the prior knowledge of location-invariant features and assuming Additive White Gaussian Noise (AWGN) channels, this design choice seems reasonable. In our experiments, all layers of the Rx module have width N_{w}. A larger layer width N_{w} is equivalent to more computing power.

The output of the Rx module can be interpreted as a representation of the image features $\mathbf{r}_i$ with index i indicating the spatial location. Thus, we have a representation of a feature map of size $(2, 2, N_{\text{w}})$ that we aggregate across the spatial dimension according to the ResNet structure. Based on this semantic representation, a softmax layer with 10 units finally computes class probabilities $q_{\boldsymbol{\varphi}}(\mathbf{z}|\mathbf{y})$ whose maximum is the maximum a posteriori estimate $\hat{z}$. In the following, we name our proposed approach Semantic INFOrmation TraNsmission and RecoverY (SINFONY).

C. Optimization Details

We evaluate SINFONY in TensorFlow 2 [40] on MNIST and CIFAR10. We split the data set into 60k/50k training data and 10k validation data samples, respectively. We do not make use of data augmentation, in contrast to [38], [39], yielding slightly worse accuracy. The ReLU layers are initialized with uniform distribution according to He and all other layers according to Glorot [41].

In the case of CIFAR10 classification with central image processing and original ResNet, we need to train $N_{\boldsymbol{\theta}} + N_{\boldsymbol{\varphi}} = 273,066$ parameters. We like to stress that although we divided the image input into four smaller pieces, this number grows more than four times to $4N_{\boldsymbol{\theta}} + N_{\boldsymbol{\varphi}} = 1,127,754$ with $N_{\text{Tx}} = N_{\text{Feat}} = 64$ for SINFONY. The reason lies in the ResNet structure with minor dependence on the input image size and that we process at four agents with an additional Tx module. Only $N_{\boldsymbol{\varphi}} = 4,810$ parameters amount to the Rx module and classification, i.e., the central unit. We note that the number of added Tx and Rx parameters of $33,560$ and $3,192$ is relatively small. Since the number of parameters only weakly grows with N_{w} in our design, we choose $N_{\text{w}} = N_{\text{Feat}}$ as the default.

For l_2-regularization, we use a weight decay of 0.0001 as in [38], [39]. For optimization of the cross entropy (10) or the loss function (18), we use the reparametrization trick from II-F and Stochastic Gradient Descent (SGD) with a momentum of 0.9 and a batch size of 64. The learning rate of 0.1 is reduced to 0.01 and 0.001 after $N_{\text{epoch}} = 100$ and 150 epochs for CIFAR10 and after 3 and 6 for MNIST. In total, we train for $N_{\text{epoch}} = 200$ epochs with CIFAR10 and for 20 with MNIST. In order to optimize the transceiver for a wider SNR range, we choose the SNR to be uniformly distributed within $[-4, 6]$ dB where $\text{SNR} = 1/\sigma_{\text{n}}^2$ with noise variance σ_{n}^2.

D. Numerical Results

In the following, we will investigate the influence of specific design choices on our semantic approach SINFONY. Then, we compare a semantic with a classical Shannon-based transmission approach. The design choices are as follows:

- **Central:** Central and joint processing of full image information by ResNet classifier, see Fig. 2. It indicates the maximum achievable accuracy.
- **SINFONY - perfect comm.:** The proposed distributed design SINFONY trained with perfect communication links and without channel encoding, i.e., Tx and Rx module, but with Tx normalization layer. Thus, the plain and power-constrained features are transmitted with $N_{\text{Tx}} = N_{\text{Feat}}$ channel uses. It serves as the benchmark since it indicates the maximum performance of the distributed design.
- **SINFONY - AWGN:** SINFONY perfect comm. evaluated with AWGN channel.
- **SINFONY - AWGN + training:** SINFONY perfect comm. trained with AWGN channel.
- **SINFONY - Tx/Rx ($N_{\text{Tx}} = N_{\text{Feat}}$):** SINFONY trained with channel encoding, i.e., Tx and Rx module, and $N_{\text{Tx}} = N_{\text{Feat}}$ channel uses.
- **SINFONY - Tx/Rx ($N_{\text{Tx}} < N_{\text{Feat}}$):** SINFONY trained with channel encoding and $N_{\text{Tx}} < N_{\text{Feat}}$ channel uses for feature compression.
- **SINFONY - Classic digital comm.:** SINFONY - perfect comm. with classic digital communications (Huffman coding, LDPC coding with belief propagation decoding, and BPSK modulation) as additional Tx and Rx modules. For details, see Sec. III-D4.
- **SINFONY - Analog "semantic" AE:** SINFONY - perfect comm. with ML-based analog communications (AE for any element in $\mathbf{r}_i$) as additional Tx and Rx modules. It is basically the semantic communication approach from [17], [22], [19]. For details, see Sec. III-D5.

Since meaning is expressed by the RV $\mathbf{z}$, we use classification accuracy to measure semantic transmission quality. For illustration in logarithmic scale, we show the opposite of accuracy in all plots, i.e., classification error rate.

1) MNIST dataset: The numerical results of our proposed approach SINFONY on MNIST are shown in Fig. 4 for $N_{\text{w}} = 56$. To obtain a fair comparison between transmit signals $\mathbf{x}_i \in \mathbb{R}^{N_{\text{Tx}}\times 1}$ of different length N_{Tx}, we normalize the SNR by the spectral efficiency or rate $\eta = N_{\text{Feat}}/N_{\text{Tx}}$. First, we observe that the classification error rate of 0.5% of the central ResNet unit with full image information (Central) is smaller than that of 0.9% of SINFONY - perfect comm.. Note that we assume ideal communication links. However, the difference seems negligible considering that the local agents only see a quarter of the full images and learn features independently based on it.

With noisy communication links (SINFONY - AWGN), the performance degrades especially for SNR < 10 dB, and we can avoid degradation just partly by training with noise (SINFONY - AWGN + training). Introducing the Tx module (SINFONY - Tx/Rx $N_{\text{Tx}} = 56$), we further improve classification accuracy at low SNR. If we encode the features from $N_{\text{Feat}} = 56$ to only $N_{\text{Tx}} = 14$ in the Tx module (SINFONY - Tx/Rx $N_{\text{Tx}} = 14$) to have less channel uses/bandwidth (stronger bottleneck), the error rate is lowest compared to other SINFONY examples with non-ideal links for low normalized SNR. At high SNR, we observe a small error offset, which indicates lossy compression. In fact, our system SINFONY learns a reliable semantic encoding to improve the classification performance of the overall system with non-ideal links. Every design choice in Tab. I is well-motivated.

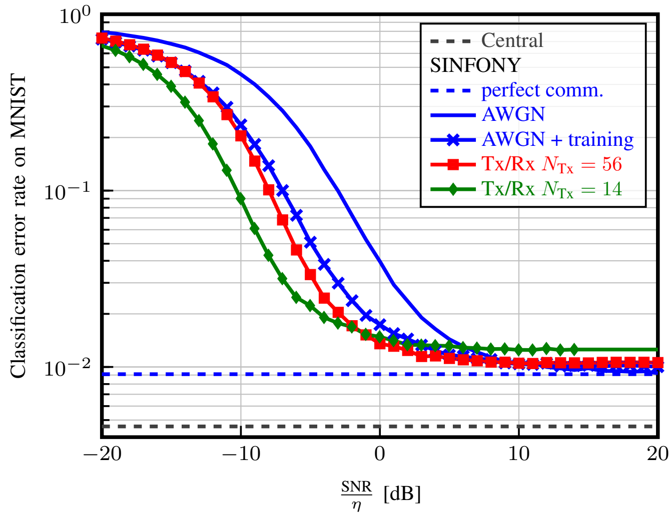

Fig. 4. Classification error rate of different SINFONY examples (distributed setting) and central image processing on MNIST as a function of normalized SNR.

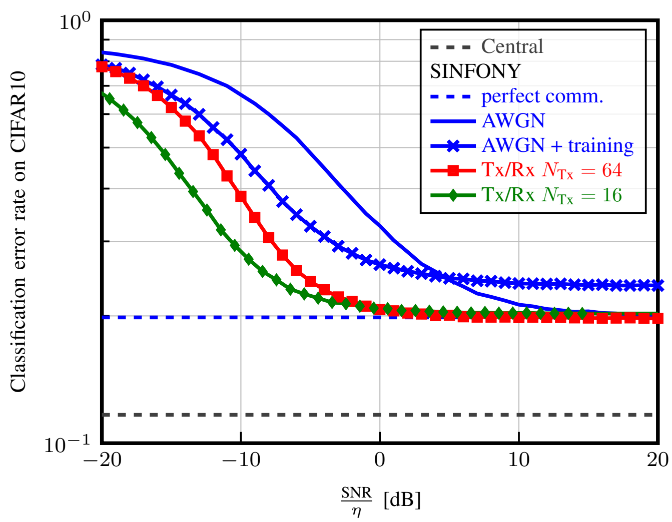

Fig. 5. Classification error rate of different SINFONY examples (distributed setting) and central image processing on CIFAR10 as a function of normalized SNR.

2) CIFAR10 dataset: Comparing these results to the classification accuracy on CIFAR10 shown in Fig. 5, we observe a similar behavior. But a few main differences become apparent: Central performs much better with 12% error rate than SINFONY - perfect comm. with 20%. We expect the reason to lie in the more challenging dataset with more

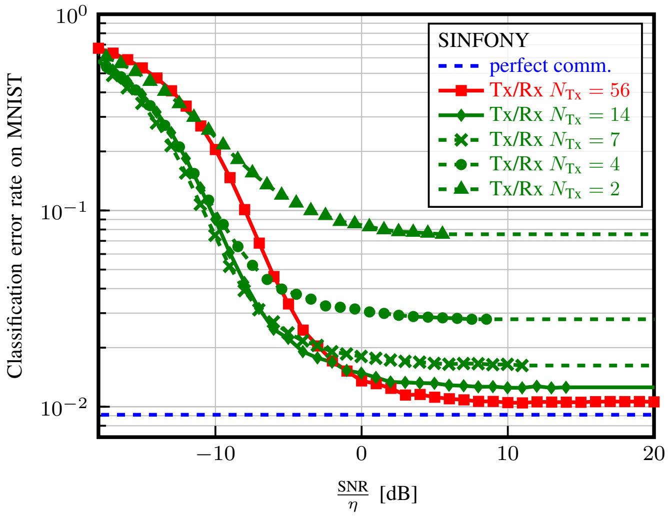

Fig. 6. Classification error rate of SINFONY for different rate/channel uses constraints as a function of normalized SNR.

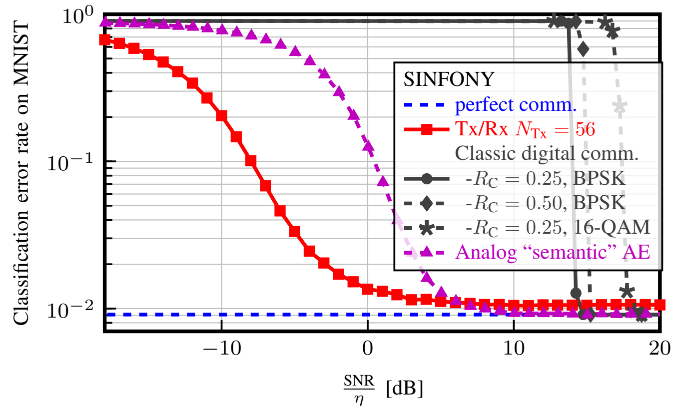

Fig. 7. Classification error rate of SINFONY with different kinds of optimized Tx/Rx modules on MNIST as a function of normalized SNR.

color channels. Further, SINFONY - AWGN + training with $N_{\mathrm{Tx}} = N_{\mathrm{Feat}} = 64$ channel uses runs into a rather high error floor. Notably, even SINFONY - Tx/Rx ($N_{\mathrm{Tx}} = 16$) with fewer channel uses performs better than both SINFONY - AWGN and SINFONY - AWGN + training over the whole SNR range and achieves channel encoding with negligible loss. This means adding more flexible channel encoding, i.e., Tx/Rx module, is crucial for CIFAR10.

3) Channel Uses Constraint: Since one of the main advantages of semantic communication lies in savings of information rate, we finally investigate the influence of the number of channel uses N_{Tx} on MNIST classification error rate shown in Fig. 6. From a practical point of view, we fix the information bottleneck by fixing the output dimension N_{Tx} and maximize the mutual information $I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})$. Decreasing the number of channel uses from $N_{\mathrm{Tx}} = 14$ to 2 and accordingly the upper bound I_{C} on the mutual information $I_{\boldsymbol{\theta}}(\mathbf{s};\mathbf{x})$, i.e., compression rate, from (17), we observe that the error floor at high SNR increases due to a larger amount of compression. For $N_{\mathrm{Tx}} = 56$, almost no error floor occurs at the cost of a smaller channel encoding gain. However, we expect that compression and channel coding are also balanced based on the channel condition, i.e., training SNR region, to find the optimal trade-off to maximize $I_{\boldsymbol{\theta}}(\mathbf{z};\mathbf{y})$, which we can also observe in unshown simulations. Following this expectation, the bound (17) is more tight in bad channel conditions, where higher compression is needed to make reliable transmission through the channel possible.

4) Semantic vs. Classic Design: Finally, we compare semantic and classic communication system designs. For the classic digital design, it would make sense to assume that the images are compressed lossless and protected by a channel code for transmission and reliable overall image classification by the central unit.

For fair comparison and ease of implementation, we instead replace Tx and Rx modules in Tab. I by a classic design: We first compress each element of the feature vector $\mathbf{r}_i$ that is computed in 32-bit floating-point precision in the distributed setting SINFONY - AWGN to 16-bit. Then, we apply Huffman encoding to a block containing 100 feature vectors of length N_{Feat}. Further, we use a 5G LDPC channel code implementation from [42] with rate $\{0.25, 0.5, 0.25\}$ and long block length of $\{15360, 16000, 15360\}$ and modulate the code bits with {BPSK, BPSK, 16-QAM}. At the receiver, we assume belief propagation decoding, where the noise variance is perfectly known for LLR computation.

The results in Fig. 7 reveal tremendous bandwidth savings for the semantic design with SINFONY: We observe an enormous SNR shift of roughly 20 dB compared to the classic digital design (SINFONY - Classic digital comm.). Note that the classic design is already near the Shannon limit and even if we improve it by ML we are only able to shift its curve by a few dB. The reason may lie in overall system optimization w.r.t. semantics and analog encoding of $\mathbf{x}$.

5) SINFONY vs. Analog "Semantic" Autoencoder: To distinguish both influences, we also implemented the approach of (14) according to Shannon by analog AEs. The analog AE has been introduced by O'Shea and Hoydis in [4]. From the point of view of semantic communication, it resembles the semantic communication approach from [17], [22], [19] without differentiating between semantic and channel coding and the mutual information constraint $I(\mathbf{x};\mathbf{y})$ like in [19]. We trained the AE with mean square error criterion for reliable transmission of any element in the feature vector $\mathbf{r}_i$ for $N_{\mathrm{epoch}} = 10000$ iterations with 10 steps per epoch and a batch size of $N_{\mathrm{b}} = 1000$, with the Adam optimizer to replace Tx/Rx modules. The Tx module per element consists of two ReLU and one linear layer with subsequent normalization of width $N_{\mathrm{Tx,el}} = 16$; the Rx module of two ReLU layers and a final linear layer providing the estimate of one element in $\mathbf{r}_i$. All intermediate, i.e., ReLU, layers at Tx/Rx module have width $2 \cdot N_{\mathrm{Tx,el}}$. With the length of the feature vector $\mathbf{z}$, we have in total $N_{\mathrm{Tx}} = N_{\mathrm{Feat}} \cdot N_{\mathrm{Tx,el}}$ and thus $\eta = N_{\mathrm{Feat}}/N_{\mathrm{Tx}} = 1/16$. We provide results (SINFONY - Analog "semantic" AE in Fig. 7): Roughly 15 dB shift is due to analog encoding. By this means, we further avoid the typical thresholding behavior of a classic digital system seen at 14 dB.

In conclusion, this surprisingly clear result justifies an analog "semantic" communications design and shows its huge potential to provide bandwidth savings. However, introducing the semantic RV $\mathbf{z}$ by SINFONY, we can further shift the curve by 5 dB compared to the analog "semantic" AE.

IV. Conclusion

Motivated by the approach of Bao, Basu et al. [13], [14] and inspired by Weaver's notion of semantic communication [2], we brought the terminus of a semantic source to the context of communications by considering its complete Markov chain. We defined the task of semantic communication in the sense of a data-reduced and reliable transmission of communications source/messages over a communication channel such that the semantic Random Variable (RV) at a recipient is best preserved. We formulated its design either as an Information Maximization or as an Information Bottleneck optimization problem covering important implementations aspects like the reparametrization trick and solved the problems approximately by minimizing the cross entropy that upper bounds the negative mutual information. With this article, we distinct from related literature [13], [14], [19], [21] in both classification and perspective of semantic communication and a different ML-based solution approach.

Finally, we proposed the ML-based semantic communication system SINFONY for a distributed multipoint scenario: SINFONY communicates the meaning behind multiple messages that are observed at different senders to a single receiver for semantic recovery. We analyzed SINFONY by processing images as an example of messages. Notably, numerical results reveal a tremendous rate-normalized SNR shift up to 20 dB compared to classically designed communication systems.

References

[1] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[2] W. Weaver, "Recent Contributions to the Mathematical Theory of Communication," in *The Mathematical Theory of Communication*, 1949, vol. 10, pp. 261–281.

[3] M. Gastpar, B. Rimoldi, and M. Vetterli, "To code, or not to code: lossy source-channel communication revisited," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1147–1158, May 2003.

[4] T. O'Shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[5] O. Simeone, "A Very Brief Introduction to Machine Learning with Applications to Communication Systems," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 4, no. 4, pp. 648–664, Dec. 2018.

[6] E. Beck, C. Bockelmann, and A. Dekorsy, "CMDNet: Learning a Probabilistic Relaxation of Discrete Variables for Soft Detection With Low Complexity," *IEEE Trans. Commun.*, vol. 69, no. 12, pp. 8214–8227, Dec. 2021.

[7] P. Popovski, O. Simeone, F. Boccardi, D. Gündüz, and O. Sahin, "Semantic-Effectiveness Filtering and Control for Post-5G Wireless Connectivity," *Journal of the Indian Institute of Science*, vol. 100, no. 2, pp. 435–443, Apr. 2020.

[8] E. Calvanese Strinati and S. Barbarossa, "6G networks: Beyond Shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107930, May 2021.

[9] Q. Lan, D. Wen, Z. Zhang, Q. Zeng, X. Chen, P. Popovski, and K. Huang, "What is Semantic Communication? A View on Conveying Meaning in the Era of Machine Intelligence," *Journal of Communications and Information Networks*, vol. 6, no. 4, pp. 336–371, Dec. 2021.

[10] E. Uysal, O. Kaya, A. Ephremides, J. Gross, M. Codreanu, P. Popovski, M. Assaad, G. Liva, A. Munari, B. Soret, T. Soleymani, and K. H. Johansson, "Semantic Communications in Networked Systems: A Data Significance Perspective," *IEEE/ACM Trans. Netw.*, vol. 36, no. 4, pp. 233–240, Jul. 2022.

[11] L. Floridi, "Philosophical Conceptions of Information," in *Formal Theories of Information: From Shannon to Semantic Information Theory and General Concepts of Information*, ser. Lecture Notes in Computer Science, 2009, pp. 13–53.

[12] W. Hofkirchner, *Emergent Information: A Unified Theory of Information Framework*. World Scientific, 2013, vol. 3.

[13] J. Bao, P. Basu, M. Dean, C. Partridge, A. Swami, W. Leland, and J. A. Hendler, "Towards a theory of semantic communication," in *2011 IEEE Network Science Workshop (NSW)*, West Point, NY, USA, Jun. 2011, pp. 110–117.

[14] P. Basu, J. Bao, M. Dean, and J. Hendler, "Preserving quality of information by using semantic relationships," *Pervasive and Mobile Computing*, vol. 11, pp. 188–202, Apr. 2014.

[15] R. Carnap and Y. Bar-Hillel, "AN OUTLINE OF A THEORY OF SEMANTIC INFORMATION," *Research Laboratory of Electronics, Massachusetts Institute of Technology*, p. 54, 1952.

[16] B. Güler, A. Yener, and A. Swami, "The Semantic Communication Game," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 4, no. 4, pp. 787–802, Dec. 2018.

[17] N. Farsad, M. Rao, and A. Goldsmith, "Deep Learning for Joint Source-Channel Coding of Text," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Canada, Apr. 2018, pp. 2326–2330.

[18] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep Learning based Semantic Communications: An Initial Investigation," in *2020 IEEE Global Communications Conference (GLOBECOM)*, Tapei, Taiwan, Dec. 2020, pp. 1–6.

[19] ——, "Deep Learning Enabled Semantic Communication Systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.

[20] Z. Weng, Z. Qin, and G. Y. Li, "Semantic Communications for Speech Signals," in *2021 IEEE International Conference on Communications (ICC)*, Virtual Conference, Jun. 2021, pp. 1–6.

[21] J. Shao, Y. Mao, and J. Zhang, "Learning Task-Oriented Communication for Edge Inference: An Information Bottleneck Approach," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 197–211, Jan. 2022.

[22] E. Bourtsoulatze, D. B. Kurka, and D. Gündüz, "Deep Joint Source-Channel Coding for Wireless Image Transmission," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 567–579, Sep. 2019.

[23] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," in *37th Annual Allerton Conference on Communication, Control and Computing*, Sep. 1999, pp. 368–377.

[24] S. Hassanpour, T. Monsees, D. Wübben, and A. Dekorsy, "Forward-Aware Information Bottleneck-Based Vector Quantization for Noisy Channels," *IEEE Trans. Commun.*, vol. 68, no. 12, pp. 7911–7926, 2020.

[25] S. Hassanpour, D. Wübben, and A. Dekorsy, "Forward-Aware Information Bottleneck-Based Vector Quantization: Multiterminal Extensions for Parallel and Successive Retrieval," *IEEE Trans. Commun.*, vol. 69, no. 10, pp. 6633–6646, Jul. 2021.

[26] O. Simeone, "A Brief Introduction to Machine Learning for Engineers," *Foundations and Trends® in Signal Processing*, vol. 12, no. 3-4, pp. 200–431, Aug. 2018.

[27] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *Journal of Machine Learning Research*, vol. 11, no. 110, pp. 3371–3408, 2010.

[28] M. Sana and E. C. Strinati, "Learning Semantics: An Opportunity for Effective 6G Communications," in *2022 IEEE 19th Annual Consumer Communications Networking Conference (CCNC)*, Virtual Conference, Jan. 2022, pp. 631–636.

[29] N. Farsad, N. Shlezinger, A. J. Goldsmith, and Y. C. Eldar, "Data-Driven Symbol Detection Via Model-Based Machine Learning," in *2021 IEEE Statistical Signal Processing Workshop (SSP)*, Virtual Conference, Jul. 2021, pp. 571–575.

[30] Z. Goldfeld and Y. Polyanskiy, "The Information Bottleneck Problem and its Applications in Machine Learning," *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 19–38, May 2020.

[31] A. Zaidi, I. Estella-Aguerri, and S. Shamai (Shitz), "On the Information Bottleneck Problems: Models, Connections, Applications and Information Theoretic Views," *Entropy*, vol. 22, no. 2, p. 151, Feb. 2020.

[32] B. M. Kurkoski and H. Yagi, "Quantization of binary-input discrete memoryless channels," *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 4544–4552, 2014.

[33] J. Lewandowsky and G. Bauch, "Information-optimum ldpc decoders based on the information bottleneck method," *IEEE Access*, vol. 6, pp. 4054–4071, 2018.

[34] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep Variational Information Bottleneck," in *5th International Conference on Learning Representations (ICLR)*, Toulon, France, Apr. 2017, pp. 1–19.

[35] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, A. Courville, and R. D. Hjelm, "MINE: Mutual Information Neural Estimation," in *Proc. of the 35th International Conference on Machine Learning (PMLR)*, Stockholm, Sweden, Jun. 2018.

[36] J. Shao, Y. Mao, and J. Zhang, "Task-Oriented Communication for Multidevice Cooperative Edge Inference," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 73–87, Jan. 2023.

[37] I. E. Aguerri and A. Zaidi, "Distributed Variational Representation Learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 120–138, Jan. 2021.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *29th IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.

[39] ——, "Identity Mappings in Deep Residual Networks," in *European Conference on Computer Vision 2016 (ECCV)*, ser. Lecture Notes in Computer Science, Amsterdam, Netherlands, Oct. 2016, pp. 630–645.

[40] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015. [Online]. Available: https://www.tensorflow.org/

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," in *IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1026–1034.

[42] J. Hoydis, S. Cammerer, F. A. Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An Open-Source Library for Next-Generation Physical Layer Research," *arXiv:2203.11854 [cs, math]*, Mar. 2022.

Edgar Beck (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Bremen, Germany, in 2014 and 2017, respectively, where he is currently pursuing a Ph.D. degree in electrical engineering at the Department of Communications Engineering (ANT). His research interests include cognitive radio, compressive sensing, massive MIMO systems, semantic communication, and machine learning for wireless communications.

Edgar Beck was a recipient of the OHB Award for the best M.Sc. degree in Electrical Engineering and Information Technology at the University of Bremen in 2017.

Dr. Carsten Bockelmann (Member, IEEE) received Dipl.-Ing. and Ph.D. degrees in electrical engineering from the University of Bremen, Germany, in 2006 and 2012, respectively. Since 2012, he has been a Senior Research Group Leader with the University of Bremen, coordinating research activities regarding the application of compressive sensing and machine learning to communication problems. His research interests include massive machine-type communication, ultra-reliable low latency communications and industry 4.0, compressive sampling, and channel coding.

Prof. Dr. Armin Dekorsy (Senior Member, IEEE) received his B.Sc. from the University of Applied Sciences Konstanz, his M.Sc. from the University of Paderborn, and his Ph.D. from the University of Bremen in 2000, all in communications engineering. He is distinguished by eleven years of industry experience in leading research positions, notably as DMTS at Bell Labs and as Research Coordinator Europe Qualcomm, and by conducting more than 50 (inter)national research projects. He is head of the Department of Communications Engineering, founder and managing director of the Gauss-Olbers Space-Technology-Transfer-Center, and has been a board member of the Technology Center for Informatics and Information Technology (TZI) for the last 10 years. His current research focuses on distributed signal processing, compressive sensing, information bottleneck method, semantic communications, and machine learning, leading to the development of technologies for 5G/6G, industrial radio, and LEO/GEO satellite communications. He is a Senior Member of the IEEE Communications and Signal Processing Society and a member of the VDE/ITG Expert Committee "Information and System Theory". He is also co-author of the textbook "Nachrichtenübertragung, Release 6, Springer', a bestseller in the German-speaking world on communication technologies.