Collaborative Edge Inference via Semantic Grouping under Wireless Channel Constraints

Mateus P. Mota*, Mattia Merluzzi*, Emilio Calvanese Strinati*,
*CEA-Leti, Grenoble, France
Email: {mateus.pontesmota, emilio.calvanese-strinati, mattia.merluzzi}@cea.fr

***Abstract*—In this paper, we study the framework of collaborative inference, or edge ensembles. This framework enables multiple edge devices to improve classification accuracy by exchanging intermediate features rather than raw observations. However, efficient communication strategies are essential to balance accuracy and bandwidth limitations. Building upon a key-query mechanism for selective information exchange, this work extends collaborative inference by studying the impact of channel noise in feature communication, the choice of intermediate collaboration points, and the communication-accuracy trade-off across tasks. By analyzing how different collaboration points affect performance and exploring communication pruning, we show that it is possible to optimize accuracy while minimizing resource usage. We show that the intermediate collaboration approach is robust to channel errors and that the query transmission needs a higher degree of reliability than the data transmission itself.**

***Index Terms*—Collaborative inference, Semantic and goal-oriented communications, edge artificial intelligence.**

I. Introduction

Edge artificial intelligence (AI) typically involves resource-poor devices, embedded with trained machine learning (ML)/AI models, ready to output inference results on data collected from complex environments. This differentiates from inferencing in central clouds, which benefits from huge computational resource but, at the same time, experience higher delays, increased energy consumption for data transfer, and greater data exposure. The success of inference depends on the quality of the collected data and the model performance. When local data are corrupted by noise or missing information, collaboration with other devices through wireless communication can help improve performance. In this direction, collaborative edge inference involves a set of devices possessing trained ML/AI models and performing the inference task in a cooperative way, helping each other in case of corrupted and/or missing local information. These devices are able to communicate to share their knowledge, allowing to improve their performance, however at the cost of added communication overhead and delay. However, the devices can perform pure local inference in case of connectivity issues, or when collaboration is not needed.

Due to its decentralized nature, collaborative inference has several advantages including but not limited to: *i)* flexibility: with each device having the complete model, this paradigm allows inference even in the face of connectivity issues; *ii)* low latency: by using proximity communications, latency is lower than using a centralized cloud server. However, it comes with challenges, such as privacy, communication design and heterogeneity. This work explores the world of edge cooperative inference from a cross-layer perspective that entails communication, computation and application aspects, under the framework of semantic and goal-oriented communications [1], a promising paradigm towards efficiently and effectively enabling AI services in 6G.

Related works A central challenge in collaborative inference is determining which features to share and how to select collaboration partners. The problem studied in this work is similar to collaborative perception [2], however, we consider the model as fixed instead of learned. In [3], a collaborative perception framework that dynamically decides when and with whom to communicate based on a query-key handshake to generate a learned communication graph. A similar framework is used in [4] for semantic data sourcing, where an edge server broadcasts a semantic query to request relevant data from its sources for a given task. This is further extended to random access in [5], where the matching between the edge server semantic query and a device key is used to determine the transmission probability. None of these papers analyze the effect of channel noise during communication, e.g., pure noise or bit/packet-level errors/erasures.

Contribution In this paper, we assume the task is performed by each edge device, i.e., closer to the framework of [3], but analyzing it under wireless system constraints such as errors introduced by wireless communication. We study the effect of channel errors in the semantic communication graph grouping [3] and the communication design choices of this method. As such, the goals of this work involve:

- Studying the effect of the channel in the collaborative inference problem.
- Studying the best splitting point and the trade-offs involved in its choice.
- Analyzing the accuracy-communication cost trade-off.

This work is structured as follows. Section II describes the system model, Section III introduces the framework for communication grouping based on semantic matching. Finally, Section IV details the experiments performed and discusses their results. The work is then concluded in Section V.

This work was funded by the 6G-GOALS Project under the HORIZON program (no. 101139232)

II. System Model

We consider a system composed of L devices empowered with AI capabilities, in this case inference models. Each device i performs inference using a pre-trained model F on input x_i, e.g., an image collected through a camera or other modality data. Without loss of generality, this model can be split into two parts, a feature extractor F_{Enc} and a decision model F_{Dec}, as in Fig. 1, such that

$$F(x) = (F_{\text{Dec}} \circ F_{\text{Enc}})\,(x)\,. \tag{1}$$

We assume that the L devices can be clustered in G groups based on the input data they generate at a given time instant, with the devices in the same group generating the same data. As such, if devices i and j are in group g, $x_i = x_j = x_g$. Without loss of generality, it is assumed that each group is composed of a number of devices and that the groups are randomly constructed, as such the devices are not aware of the identity of other devices in their group. The ultimate goal is to mutually discover these identities to improve local inference performance, with low overhead for the network, i.e. without the need of directly sharing data.

With probability p_{p}, a device has access only to partial or noisy observation of the true data, i.e., the true group observation, x_g, given by $\hat{x}_i = M^i(x_g)$, with $i \in [1\,..\,L]$ and $g \in [1\,..\,G]$. Otherwise, with probability $1 - p_{\text{p}}$, the device has access to the true observation, x_g. All devices are able to communicate by sharing the output of the first part of the model $F_{\text{Enc}}\,(\hat{x}_i)$, which is then shared with other devices (at least one) over a wireless communication link C. We intentionally keep the notion of communication system general. The latter can be translated into a wireless link, end-to-end link to edge, or whatever transformation and transportation performed on data. As an example, in this work, C is represented by a wireless packet erasure channel. This results in a possibly corrupted version of data at the receiver.

For a device i, the information received by other devices in its group is instrumental to improve local inference performance, especially in the case of noisy or corrupted local observation. However, this requires the discovery of the devices belonging to the same group, as well as good enough channel conditions. Whenever a generic message o_j is transmitted by device j, the message y_c^{ij} received by device i, after passing through said communication system is denoted by

$$y_c^{ij} = C\,\{o_i\}\,, \tag{2}$$

where we use the notation $\{\cdot\}$ to denote a system, rather than a function.

The devices need to aggregate the information received by their created group. This is performed by the feature combiner, F^i_{Comb} at device i. Denoting by $\mathbf{y}_c^i$ the aggregated information received by device i from the other devices in the group, the output of the feature combiner is

$$y_g^i = F_{\text{Comb}}\left(F_{\text{Enc}}(\hat{x}_i), \mathbf{y}_c^i\right). \tag{3}$$

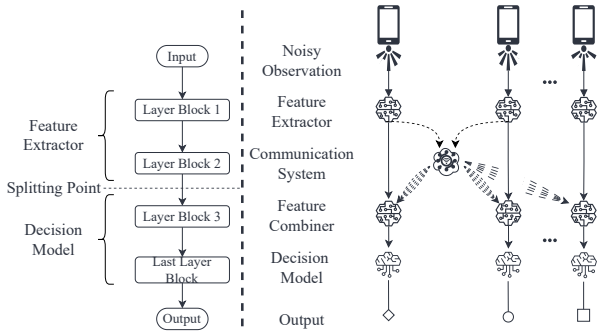


Fig. 1. System model scheme for the proposed collaborative inference problem.

Finally, the combined information is fed to the decision model to provide the inference result at device i:

$$y_d^i = F_{\text{Dec}}(y_g^i)\,. \tag{4}$$

This whole procedure is illustrated in Fig. 1.

The end goal of this setting is to allow collaborative inference, improving the overall accuracy by means of sharing intermediate information over wireless. For this reason, the devices are allowed to use the communication system, not only to share their feature information, but also to identify potential devices with the information relevant as per the scope of improving their inference task performance. This needs weighting their contribution in the feature combiner. Final performance of this collaboration depends on: (i) the effectiveness of group creation, (ii) the wireless channel quality between collaborating devices, and (iii) that the splitting point of model F. The latter not only affects performance in terms of accuracy, but also the communication system, since different splitting points provide different feature size.

A. Communication System

Sidelink communication between devices can happen in three different ways:

- **Unicast:** One-to-one communication. Data is transmitted to a single device.
- **Multicast:** One-to-many communication. Data is transmitted to a dedicated set of devices in the area.
- **Broadcast:** One-to-all communication. Data is transmitted to all devices in the broadcast area.

Unicast and multicast communication need network connection, since they use the uplink to request communication, i.e. request to join a group. The solution studied in this paper relies on both multicast for the transmission of *semantic queries* and unicast to exchange intermediate observation.

The communication system C is a sidelink packet erasure channel, where a transport block (TB) is incorrectly received with a probability defined as packet error rate (PER). The transmitted data is divided into TBs of size N, which for simplicity we assume to be the number of floating-points values. The system is assumed to have constant resources,

such that N and PER are fixed. Also, we do not assume retransmissions in this work, so that if a TB is not received then its values are filled with a default value, e.g. 0.

B. Key Performance Indicators

Semantic and goal-oriented communication goes beyond classical wireless communication metrics towards application success. Focusing on image classification, we consider accuracy as key performance indicator at the application level.

However, application performance are typically to be traded off with cost, e.g., in terms of communication and computation. Here, focusing on the sidelink, we consider the average resource usage as KPI. The latter is computed as the average resource usage of the system, namely the number of sidelink transmissions resulting from the optimized device grouping (i.e., the communication graph).

III. Semantic matching-based grouping

Given the above task, this work leverages an attention-based mechanism similar to [3] to identify: i) whether a device needs extra information for inference due to corrupted local data, and ii) the set of devices to collaborate with, in case of bad quality local data. Adapting the framework to our system, device i compresses its observation, obtaining an intermediate representation $o_i = F_{\text{Enc}}(\hat{x}_i)$. Then, it generates: *i)* a low-dimensional query vector μ_i and *ii)* a key vector κ_i:

$$\mu_i = Q(o_i; \theta_q), \tag{5}$$

$$\kappa_i = \mathcal{K}(o_i; \theta_k), \tag{6}$$

where $\mathcal{K}$ and Q are two neural networks parametrized by θ_k and θ_q, respectively. The query is transmitted to all other devices, while the key is kept local. The query received by device j from device i is $\hat{\mu}_i^j = C\{\mu_i\}$.

All devices receive the queries of all others (multicast), and uses their keys to compute a matching score through scaled general attention [6]. Then, it exchanges data (unicast). We denote by m_{ij} the matching score for device j receiving query from device i, which reads as

$$m_{ij} = \frac{\kappa_i{}^{\intercal} \mathbf{W} \hat{\mu}_i^j}{\sqrt{K}}, \tag{7}$$

where $\mathbf{W} \in \mathbb{R}^{Q \times K}$ is a learnable parameter to match the query size, Q, and the key size, K.

All the matching scores are used to construct a matching matrix $\mathbf{M}$ by using a row-wise softmax, with elements $\bar{m}_{ij}$. The latter is used to construct the communications graph, as its values $\bar{m}_{ij}$ represent how relevant the information of device i is for device j. Once the groups are created, the devices share the actual data (or, their intermediate representation), which might have high dimensionality compared to the queries. To avoid high communication overhead, $\mathbf{M}$ can be pruned with threshold ρ, i.e., $\bar{m}_{ij}^{\rho} = \bar{m}_{ij} \cdot \mathbf{1}\{\bar{m}_{ij} \geq \rho\}$, where $\mathbf{1}\{\cdot\}$ denotes the indicator function. $\mathbf{M}$ is also used to combine features (cf. (3)) according to the following weighted average:

$$y_g^i = \sum_{j=1}^{L} \bar{m}_{ij}^{\rho} y_c^{ij}. \tag{8}$$

where y_c^{ij} is the received version of the intermediate data, as per the definition in (2).

In this work, we consider image classification as application. As such, training is performed by computing the cross-entropy loss between the true label and the predicted label, $y_d^i = F_{\text{Dec}}(y_g^i)$. It is important to highlight that, differently from [3], only the query generator Q, the key generator $\mathcal{K}$ and the attention weights $\mathbf{W}$ are learned. As such, the encoder model F_{Enc} and the decoder model F_{Dec} are assumed to be pre-trained and their parameters frozen, while only the modules needed for the communication need to be trained. As a consequence, the learned encoder and decoder models are shared across all devices. Note that decentralized training is also possible, with the result of different models for each device. However, this increases the computational cost. Differently from [5], training to learn key and query takes into consideration the channel effects, i.e. packet losses, instead of only considering it during execution. This introduces robustness to noise and thus the possibility of improving inference performance through wireless collaboration, without the need for high communication reliability, e.g., through retransmissions.

IV. Experiments and Discussion

A. Simulation details and parameters

Image classification is performed on the Imagenette dataset [7]. The pre-trained model is the MobileNetV3-Small [8], initialized with its default weights from training in the ImageNet dataset and then fine-tuned to the Imagenette dataset. The partial observability is modeled by applying a white patch in a random position of the image, with the ratio between the white patch size and the image size being 0.4. In other words, 40% of the image is locally missing at the device, if the latter belongs to the set of devices with corrupted data. We do not consider generalization in the following results, as the system is trained for each combination of parameters considered.

Unless otherwise stated, it is assumed that the query vector is transmitted reliably, in an error-free manner and given its size compared with the intermediate feature vector, its contribution to the resource usage is not neglected. The intermediate feature vector is thus the only information affected by channel errors.

We highlight that computation overhead added by the key-query and attention modules is negligible with respect to the base model (encoder and decoder). With the architectures used, the total number of flops for running MobileNetV3-Small is 55M multiply-accumulate operations (MACs), while the key and query networks amount to only 1.25M and 1.37M MACs, respectively, and the attention module to 0.07M MACs.

The semantic grouping solution is compared with two other benchmark solutions:

- **Local inference:** only the local observation is used for inference, possibly on corrupted data. This represents the non-collaborative case.
- **Noiseless:** collaboration is performed without wireless channel impairments. This represents a performance upper bound

TABLE I
SIMULATION PARAMETERS

Parameter	Symbol	Value
Number of devices	L	16
Number of groups	G	4
Packet error rate	PER	10^{-1}
Transport block size (# of floating-points)	L_{TB}	40
Filling value for packet losses		0.0
Query size	Q	64
Key size	K	1024
Patch scale		0.4
Probability of partial observation	p_{p}	0.8

TABLE II
TRAINING PARAMETERS

Parameter	Value
Batch size	64
Number hidden layers	2
# of neurons per hidden layer in $\mathcal{K}$ and Q	[256, 128]
Activation function of hidden neurons	ReLU
Optimizer algorithm	Rectified Adam
# of epochs	60
Learning rate	10^{-5}

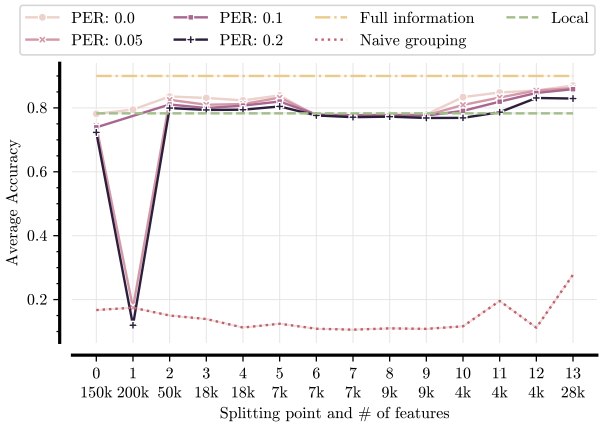


Fig. 2. Accuracy and # of features when varying the splitting point.

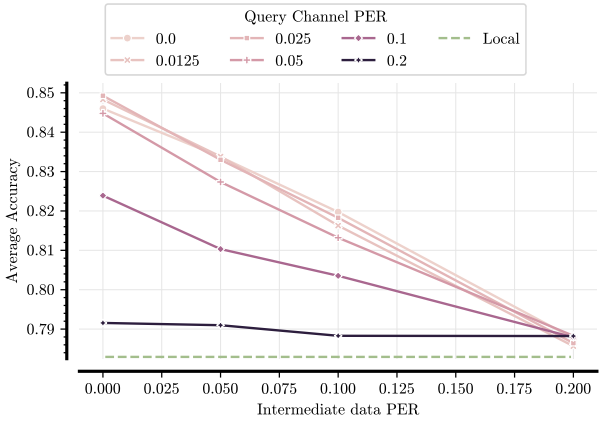


Fig. 3. Accuracy for different query and intermediate data PERs. itting point: 11.

- **Naive grouping:** the observation of all devices are averaged with the same weights, not based on the semantic data relevance.

We analyze numerical results by varying: i) the packet error rate, ii) the DNN splitting point for extracting data for collaboration, iii) pruning threshold, and ultimately the channel errors on the queries.

1) Splitting point choice: First, we analyze the effect of the splitting point choice, comparing the semantic grouping solution trained with different PERs with the baselines in Fig. 2. Namely, we show the accuracy as a function of the splitting point, for different PERs. The size of the intermediate feature observation is highlighted below each splitting point in the abscissa.

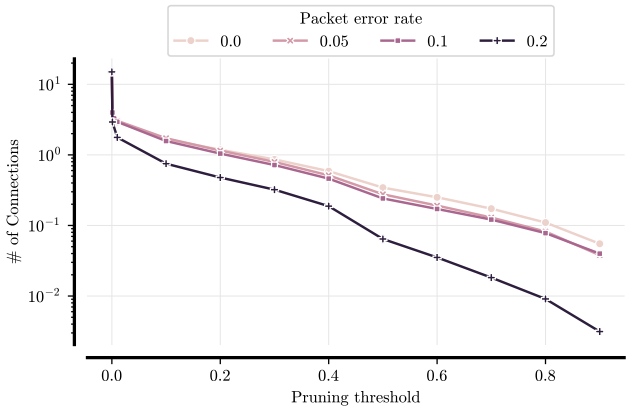
From Fig. 2, we can draw the following conclusions:

- Robustness to noise depends splitting point, as initially observed in [9]: Some splitting points (e.g., 4 and 12) show a strong robustness, where there is little performance variation, while other points are more susceptible to channel errors, such as points 0 and 1;
- Naively grouping devices leads to extremely poor performance, as it does not take into account the semantic relevance of data when collaborating;
- It is possible to improve accuracy through collaboration while transmitting less information, as shown by the increased performance in terms of accuracy when going deeper before collaborating. This helps reducing the resource usage per transmission.

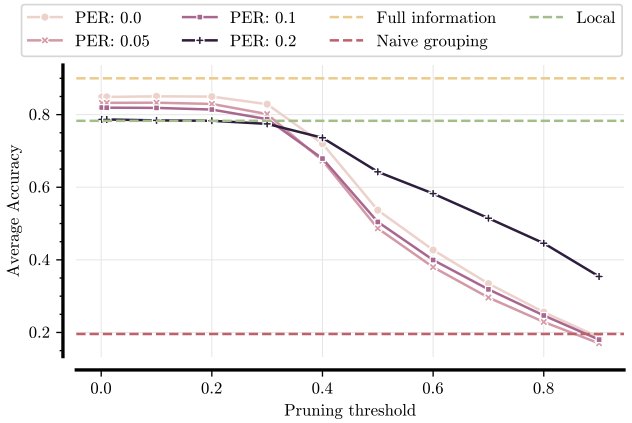
The above conclusions imply that the physical layer (PHY) can relax its reliability levels as long as an appropriate semantic extraction can be performed. We also note that naive grouping does not perform well since it fails to filter the data based on its relevance to a device.

2) Channel effect on query and data transmission: We conclude that PHY reliability is not necessary when semantic information is exchanged. However, in the previous result, we considered error-free communication for the query. In Fig. 3, we plot the accuracy as a function of the PERs during intermediate data transmission, for different PERs during query transmission. The system is trained for each combination of query and data channel PERs.

We can notice the severe effect of query errors, as the accuracy drops nearly twice as much with an increase in the query channel PER compared to a similar increase in the intermediate data channel PER. This suggests that the query needs a reliable transmission scheme, compared to the data (or intermediate representation) itself. We can conclude that semantic representation helps communication robustness, but device grouping needs reliable query exchange to achieve acceptable performance. However, it should be noted that, given the small size of the query, increased communication reliability effort does not impact system cost as data transmission itself.



(a) Effect of the pruning threshold on the average number of sidelink connections per device.



(b) Effect of the pruning threshold on the average accuracy.

Fig. 4. Studying the effect of the communication pruning. The lowest thresholds are $[0, 0.001, 0.01]$. Splitting Point: 11

3) Communication pruning effect: We now analyze the effect of the communication pruning, which reduces the communication by only transmitting information if the matching score is above a certain threshold. These results are shown in Fig. 4. In Fig. 4a, we plot the average number of sidelink connections per inference task, as a function of the pruning threshold ρ. Whereas, Fig. 4b shows the corresponding accuracy, again as a function of ρ. Fig. 4a shows that the lowest non-negative threshold already provides significant communication reduction. This is thanks to the fact that some of the elements of the matching matrix are very close or equal to zero. Naturally, this does not reduce accuracy, as shown in Fig. 4b, because only lowly weighted collaborations are pruned. Fig. 4a also shows that bigger pruning reduces communication even more when the channel conditions worsen. This implies that the matching scores have lower variation under strong channel impairments, which insinuates that the system learns to rely on information from more sources when PER is high than when it is low, in which case the system instead relies on few relevant sources. Comparing Fig. 4a and Fig. 4b, we can conclude that it is possible to reduce communication effectively without affecting accuracy. However, if communication is heavily reduced, the degradation in performance can even overcome local performance with corrupted data.

V. Conclusions and Perspectives

We investigated the impact of wireless communication impairments on collaborative inference in edge AI systems, focusing on semantic grouping, communication efficiency, and model partitioning under varying channel conditions. By leveraging a key-query mechanism for selective feature exchange, we demonstrated that adaptive communication strategies can significantly improve inference accuracy while minimizing resource usage. We showed that the semantic grouping solution is robust to channel errors. Nevertheless, the query channel requires more reliability than the data channel. Furthermore, it is possible to improve the accuracy of the inference task while reducing the communication cost by appropriate choice of the splitting point and communication graph pruning.

Our findings provide practical guidelines for designing scalable and communication-aware edge AI deployments and operations. Future research directions include extending the framework to consider the channel information in the matching matrix, so that the model can handle different channel conditions. Another interesting perspective is a query-aware data transmission, such that the query is used to extract the more relevant features to be transmitted. In terms of evaluation, a comparison on different tasks such as data sourcing [4] and semantic segmentation [3] can provide better insights into the wireless effects across different tasks.

References

[1] E. C. Strinati *et al.*, "Goal-oriented and semantic communication in 6g ai-native networks: The 6g-goals approach," in *2024 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 2024, pp. 1–6.

[2] Y. Han, H. Zhang, H. Li, Y. Jin, C. Lang, and Y. Li, "Collaborative perception in autonomous driving: Methods, datasets, and challenges," *IEEE Intelligent Transportation Systems Magazine*, vol. 15, no. 6, pp. 131–151, 2023.

[3] Y.-C. Liu, J. Tian, N. Glaser, and Z. Kira, "When2com: Multi-agent perception via communication graph grouping," in *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2020, pp. 4106–4115.

[4] K. Huang, Q. Lan, Z. Liu, and L. Yang, "Semantic data sourcing for 6g edge intelligence," *IEEE Communications Magazine*, vol. 61, no. 12, pp. 70–76, 2023.

[5] A. E. Kalør, P. Popovski, and K. Huang, "Random access protocols for correlated iot traffic activated by semantic queries," in *2023 21st International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*. IEEE, 2023, pp. 643–650.

[6] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[7] J. Howard, "Imagenette: A smaller subset of 10 easily classified classes from imagenet," March 2019. [Online]. Available: https://github.com/fastai/imagenette

[8] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.

[9] F. Binucci, M. Merluzzi, P. Banelli, E. C. Strinati, and P. Di Lorenzo, "Enabling edge artificial intelligence via goal-oriented deep neural network splitting," in *2024 19th International Symposium on Wireless Communication Systems (ISWCS)*, 2024, pp. 1–6.