Neurocomputing 572 (2024) 127132

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Evidence, my Dear Watson: Abstractive dialogue summarization on learnable relevant utterances

Paolo Italiani[1], Giacomo Frisoni[1,*], Gianluca Moro[1], Antonella Carbonaro, Claudio Sartori

Department of Computer Science and Engineering (DISI), University of Bologna, Via dell'Università 50, Cesena (FC), I-47522, Italy

ARTICLE INFO

Communicated by V. Garcia-Diaz

Keywords:
Abstractive dialogue summarization
Input augmentation
Text classification
Gumbel-softmax trick
Interpretable natural language processing

ABSTRACT

Abstractive dialogue summarization requires distilling and rephrasing key information from noisy multi-speaker documents. Combining pre-trained language models with input augmentation techniques has recently led to significant research progress. However, existing solutions still struggle to select relevant chat segments, primarily relying on open-domain and unsupervised annotators not tailored to the actual needs of the summarization task. In this paper, we propose DearWatson, a task-aware utterance-level annotation framework for improving the effectiveness and interpretability of pre-trained dialogue summarization models. Precisely, we learn relevant utterances in the source document and mark them with special tags, that then act as supporting evidence for the generated summary. Quantitative experiments are conducted on two datasets made up of real-life messenger conversations. The results show that DearWatson allows model attention to focus on salient tokens, achieving new state-of-the-art results in three evaluation metrics, including semantic and factuality measures. Human evaluation proves the superiority of our solution in semantic consistency and recall. Finally, extensive ablation studies confirm each module's importance, also exploring different annotation strategies and parameter-efficient fine-tuning of large generative language models.

1. Introduction

In today's fast-paced world, dialogues have become ubiquitous and indispensable means of communication, with online chat applications (e.g., Whatsapp, Messenger, WeChat) being the most prominent case [1]. Automatic highlights can aid individuals and organizations in managing and comprehending the increasingly overwhelming amount of information exchange, ultimately supporting quick reviews and decision-making.

In the natural language generation (NLG) field, abstractive dialogue summarization is the task of producing a condensed and meaningful summary of a multi-speaker conversation that is not a verbatim representation of the original input. Compared to traditional writings, human-to-human dialogues pose unique challenges due to their dynamic, interactive, and first-person nature, often informal, prolix, and repetitive—peppered with false starts, backchanneling, speaker role shifting, reconfirmations, hesitations, and interruptions [2]. Relevant content is often scattered across participants and dialogue turns, resulting in a lower information density and more diffuse topic coverage. These obstacles become even more pronounced when style and register are diversified, the setting is multi-party (>2 interlocutors), and sources reflect heterogeneous real-life subjects. Despite the recent strides achieved by pre-trained language models (PLMs), existing summarizers have *opaque behaviors* and pay attention to information pieces different from those included in the reference summary [3], undermining trustability.

The latest research has focused on input augmentation techniques to guide the generative model in identifying the most fundamental concepts, incorporating auxiliary annotations into the dialogue text. Prior studies have sought to automatically annotate keywords, redundant utterances, and topics [4]. Nevertheless, these annotations usually come from *burdensome human efforts* [5], *open-domain toolkits* [6]—not suitable for dialogues, or *unsupervised strategies* [7–10]—not designed to complement the task of interest, resulting in possible inconsistencies and information loss. The annotation process (gold or silver) is treated as a preprocessing step rather than a learning objective, capping potential benefits and generalization. Furthermore, Srivastava et al. [11] have recently demonstrated the advantages of training or making zero-shot summary predictions exclusively with pertinent source sentences determined through fixed algorithms grounded on ROUGE-1 and topic segmentation. These advances raise expectations

* Corresponding author.
E-mail addresses: paolo.italiani@unibo.it (P. Italiani), giacomo.frisoni@unibo.it (G. Frisoni), gianluca.moro@unibo.it (G. Moro), antonella.carbonaro@unibo.it (A. Carbonaro), claudio.sartori@unibo.it (C. Sartori).
[1] Equal Contribution.

https://doi.org/10.1016/j.neucom.2023.127132
Received 19 June 2023; Received in revised form 13 October 2023; Accepted 6 December 2023
Available online 15 December 2023

for learnable, reference-free, and dynamic annotation techniques, still lacking contributions.

We propose DearWatson, a novel task-aware utterance-level annotation framework designed to enhance the efficacy and interpretability of neural dialogue summarizers. Drawing on the knowledge encoded in PLMs, we train a classifier to detect the most relevant dialogue utterances for the summarization task. Predicted utterances are then wrapped with special tokens, and a pre-trained summarizer is fine-tuned on the augmented input. As depicted in Fig. 1, we postulate that pointing out the meaningful source spans allows the generative model to better direct its attention to them; learning such annotations together with the summary can enhance final performance and unlock interpretability. We investigate two architectures following different learning schemes: (i) *joint*, where the annotator and the summarizer are treated as two independent units simultaneously trained, and the relevant target utterances come from a self-supervised heuristic; (ii) *end-to-end*, where the gradient backpropagates from the summarizer to the annotator, whose tag placement directly optimizes summary generation. Differentiable utterance selection is achieved via Gumbel-Softmax Trick [12]. We run extensive experiments on two widely used dialogue summarization datasets, SAMSum [3] and DialogSum [13], demonstrating that our models push the state-of-the-art according to five automatic evaluation metrics. Through ablation studies, we establish the significance of each module, ranging from the annotator implementation and underlying heuristic to the influence of the number of speakers and utterances. Qualitative experiments show a less diluted distribution of attention scores. Furthermore, human and ChatGPT-driven evaluations corroborate the evidence role of the selected utterances in making the summarizer behavior more transparent.

Our contributions can be summarized as follows:

- We pioneer input augmentation (summary-worthy utterance selection) as an additional training objective, without any reliance on human-crafted labels.
- We shed light on the overall efficacy of input augmentation in dialogue summarization, evaluating different annotation heuristics, and uncovering substantial room for future improvements.
- Beyond joint learning, we propose an architecture capable of making discrete choices without disrupting backpropagation. To this end, we suggest the incorporation of Gumbel-based sampling algorithms, a methodology previously unexplored in the context of evidence generation [14]. Our research offers valuable insight into the intricate comparison between joint learning and end-to-end learning.
- We outperform all existing summarization baselines on two popular testbeds (up to +0.67/0.58/0.63 ROUGE-1/2/L F1 and +1.38 BERTScore, i.e., maximum Δ for each metric regardless of the dataset), and add interpretability with only minor incremental training/inference costs associated with the extra tagging task.
- We carry out a rigorous human evaluation to demonstrate that learned annotations are instrumental in understanding the dialogue segments that exert the greatest impact on a model prediction.
- We mark the first instance of harnessing ChatGPT for abstractive dialogue summarization, gauging its validity in terms of consistency with human judgment.

2. Related work

Dialogue summarization. Pre-trained language models have sparked a paradigm shift in abstractive summarization [15,16], yielding unprecedented results even in multi-document [17,18] and low-resource [19–21] settings. Although most prior work addresses single-speaker content, dialogue summarization is gaining traction. Nevertheless, transferring conventional models to chats is not straightforward: avoiding overlooking essential information across turns is one of the primary requirements [22]. Since core knowledge is often fragmented and embodied in incomplete sentences, generating fluent summaries by utterance extraction alone is impractical. Popular approaches involve ad-hoc pretraining [23–29], while – following the steps of current trends on graph signals [30–32] – a complementary procedure is injecting structured data modeling conversation topics or dialogue acts [9,33]. In this paper, we enrich the dialogue text to enhance the effectiveness and interpretability of any summarizer.

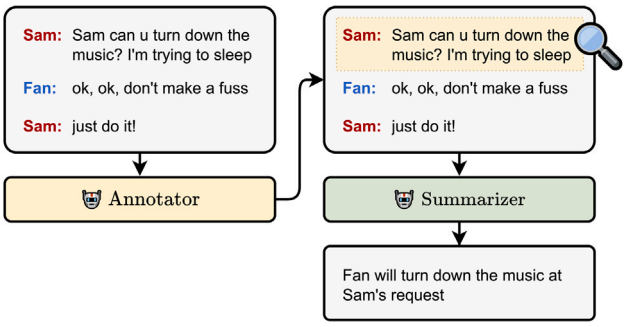


Fig. 1. Overview of DearWatson. Utterance annotation and summarization are simultaneously mastered.

Annotation for abstractive summarization. Conceptually, Peyrard [34] posits that a high-quality summary is closely tied to three dimensions: informativeness, redundancy, and relevance. Past research has actively accounted for these aspects by incorporating auxiliary information into the dialogue. To bolster informativeness, some work tagged linguistically-grounded keyphrases [35], domain terminologies [36], and topic words [37]. To mitigate redundancy, the authors labeled repetitive utterances with similarity-based methods [6,7]. To maximize relevance, other groups carried out topic segmentation [5,8,9]. Feng et al. [10] drawn support from forward passing on a pre-trained conversational response generation model to tag the input text based on loss scores and utterance embeddings. Input augmentation has also been exploited to control text generation, entailing the introduction of auxiliary signals to enforce the desired output properties [22,38,39]. All such annotations are procured through manual labor, predefined heuristics, open-domain toolkits, or unsupervised techniques. Instead, we conjecture that model performance remains subpar when treating input augmentation outside of learning objectives. To our knowledge, we are the first to optimize the annotator in tandem with the summarization task.

3. Preliminaries

3.1. Task definition

Abstractive dialogue summarization endeavors to craft a creative summary S of $|S|$ tokens $[s_1, s_2, \dots, s_{|S|}]$ from a dialogue D consisting of $|D|$ utterances $[\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{|D|}]$. $[\cdot]$ denotes concatenation. Each utterance $\mathbf{u}_i$ comprises $|\mathbf{u}_i|$ tokens $[\mathbf{p}_i, u_{i,1}, u_{i,2}, \dots, u_{i,|\mathbf{u}_i|}, \texttt{SEP}_i]$, where $i \in [1, |D|]$, $\mathbf{p}_i$ is the speaker, and $\texttt{SEP}_i$ signifies the utterance end. Therefore, the task can be formalized as producing the summary S given the input: $\mathcal{D} = [\mathbf{p}_1, u_{1,1}, \dots, \texttt{SEP}_1, \dots, \mathbf{p}_{|D|}, u_{|D|,1}, \dots, \texttt{SEP}_{|D|}]$.

3.2. Oracle dialogue annotation

Chat data is rife with irrelevant content, such as greetings and inconsequential turns, that does not contribute to producing an informative resume. Hence, steering the model's attention toward core utterances may bear substantial opportunities. Gold utterance relevance labels for a dialogue summarization task are generally not available, and acquiring them would be a costly endeavor. To make our solution adaptable

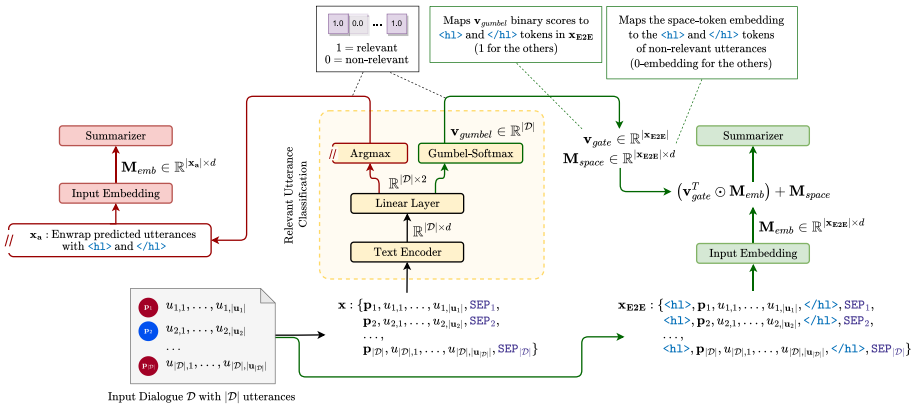


Fig. 2. Illustration of our joint and end-to-end DearWatson architectures. Following a shared initial phase for utterance classification, joint learning peculiarities are highlighted in red, while the alternative end-to-end modifications are shown in green. The "//" symbol connotes backpropagation interruption points.

and effective in real-world scenarios, we have devised multiple self-supervised heuristics. Taking a closer look, we propose three alternative approaches to designate a source utterance $\mathbf{u}_i$ as relevant based on its *overlap* ($\cap$) with the gold summary reference.

- One-word-overlap: inspired by Wu et al. [38], $\mathbf{u}_i$ must feature at least one word (other than stopwords) from the target summary.
- Top-p-word-overlap: more severely, $\mathbf{u}_i$ must rank within the top $p\%$ of the source utterances that exhibit the highest similarity to the target summary, measured using ROUGE-1.
- Top-p-semantic-overlap: ——"—— measured using cosine similarity as in [40,41]. To accomplish this, we leverage a state-of-the-art Sentence Transformer,[2] computing the normalized dot product between semantically-informed utterance embeddings.

Please note that, unlike One-word-overlap, he hyperparameter p offers control over the annotation percentage. According to the relevance criterion under investigation, we flag core utterances by inserting the special tokens <hl> and <\hl> at their beginning and end, respectively. Eq. (1) succinctly outlines the utterance map function for producing the annotated dialogue $\mathcal{D}^*$.

$$\text{ann}(\mathbf{u}_i) = \begin{cases} [\langle \texttt{hl} \rangle_i, \mathbf{p}_i, \ldots, \langle \texttt{\hl} \rangle_i, \text{SEP}_i] & \mathbf{u}_i \cap \mathcal{S} \neq \emptyset \\ \mathbf{u}_i & \mathbf{u}_i \cap \mathcal{S} = \emptyset. \end{cases} \tag{1}$$

We note that this is only possible during training, since we obviously do not have access to the target summary at inference time. Empirical tests from our group exhibit that PLMs fine-tuned on oracle $\mathcal{D}^*$, rather than $\mathcal{D}$, gain up to 4.5 ROUGE-1/2/L average points (see Section 6.1 for details). These results attest to the great controllability power of the utterance annotation strategy and motivate the architectures discussed in Section 4, where we naturally eliminate the prior target knowledge assumption to define general-purpose *annotate-then-summarize* models.

4. Method

This section will introduce two supervised architectures for highlighting relevant utterances and condensing the augmented dialogue. First, we will explore a *joint* paradigm, breaking down the goal into two subtasks learned simultaneously (Section 4.1). Then we will move to an *end-to-end* framework, taking annotation and summarization as a single unit by differentiable input augmentation (Section 4.2). We sketch both solutions in Fig. 2.

[2] https://huggingface.co/sentence-transformers.

4.1. Joint learning paradigm

4.1.1. Utterance classification

We fine-tune a PLM to predict relevant utterances, i.e., dialogue lines mentioning facts expected to appear in the resume. To achieve this goal, the ann($\mathbf{u}_i$) heuristic disclosed in Section 3.2 is treated as the source of ground-truth annotations. Technically, we supply $\mathcal{D}$ to a bidirectional text encoder $E_t(\cdot)$, deriving a contextual hidden representation $\mathbb{R}^d$ for each dialogue token:

$$[\mathbf{h}_{\mathbf{p}_1}, \mathbf{h}_{u_{1,1}}, \ldots, \mathbf{h}_{\text{SEP}_{|\mathcal{D}|}}] = E_t(\mathcal{D}). \tag{2}$$

We extend the model training to represent each utterance in the corresponding SEP token. Indeed, on top of the encoder stack, we add a task-specific linear layer to project each utterance embedding $\mathbf{h}_{\text{SEP}_i}$ into two-dimensional unnormalized logits $\mathbf{l}_i$ (confidences for relevant T and non-relevant F classes):

$$[\mathbf{l}_1, \ldots, \mathbf{l}_{|\mathcal{D}|}] = \mathbf{W}_c\{\mathbf{h}_{\text{SEP}_1}, \ldots, \mathbf{h}_{\text{SEP}_{|\mathcal{D}|}}\} + \mathbf{b}_c, \tag{3}$$

where $\mathbf{W}_c$ represent trainable weights and $\mathbf{b}_c$ are bias parameters. We optimize a binary cross-entropy loss $\mathcal{L}_{ce}$:

$$\mathcal{L}_{ce} = -\sum_{i}^{\{T,F\}} y_i \log(p_i), \tag{4}$$

where y_i is the truth class label and p_i is the softmax probability on the output of Eq. (3) for the ith class.

4.1.2. Summarization

We unveil relevant utterances by applying the argmax function to the output of the classification head, wrapping them with special <hl><\hl> tokens. Finally, we feed the annotated dialogue $\mathcal{D}^*$ to a pre-trained transformer-based summarizer. To maximize the estimated probability P_θ of the actual summary $\mathcal{S}$, we utilize a negative log-likelihood loss function:

$$\mathcal{L}_{nll} = -\sum_{i} \log P_\theta(s_i | \mathcal{D}^*). \tag{5}$$

The classifier and the summarizer are jointly trained by minimizing $\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{nll}$.

4.2. End-to-end paradigm

The utterance annotation process described in Section 4.1 halts the gradient flow from the summarizer to the classifier. Non-differentiability is caused by the argmax step function, which outputs discrete class labels, i.e., hard assignments. When concerned with discrete stochasticity, challenges arise regarding both sampling from discrete distributions and gradient estimation thereof. Reinforcement

learning-free approaches often relax sampling functions by employing continuous approximations, such as Soft-argmax [42] or Gumbel-Softmax [43]. Gumbel-based sampling algorithms, in particular, have found success in a range of applications, such as generating text through stochastic beam search for dialogue systems [44] and machine translation [45–47], as well as leaning communication protocols in multi-agent games [48]. In this paper, our proposed end-to-end architecture incorporates the Gumbel-Softmax Trick for categorical reparameterization.

Gumbel-softmax trick. Gumbel-Softmax is an efficient gradient estimator for sampling from a categorical distribution, which unlocks backpropagation through the entire network even when intermediate outputs are discrete—directly mapping the raw dialogue to the summary on the augmented input. Let c be our categorical variable with probabilities $\{\pi_T, \pi_F\}$. Categorical samples are encoded as two-dimensional one-hot vectors at the corners of a mono-dimensional simplex Δ^1. Utterance relevance choices from our $|D|$-length c distribution are estimated by generating $z \in \Delta^1$:

$$z_i = \frac{\exp\left(\left(\log\left(\pi_i\right) + g_i\right)/\tau\right)}{\sum_j^{\{T,F\}} \exp\left(\left(\log\left(\pi_j\right) + g_j\right)/\tau\right)} \quad \text{for } i \in \{T, F\}, \tag{6}$$

where τ is the softmax temperature and g_T, g_F are i.i.d. noise samples drawn from Gumbel$(0, 1)$ by pulling out $u \sim$ Uniform$(0, 1)$ and then computing $g = -\log(-\log(u))$. At lower temperatures ($\tau \to 0$), z samples are identical to those from a categorical distribution; at higher temperatures ($\tau \to \infty$), z samples are no longer one-hot and become uniform. The Gumbel-Softmax distribution is smooth for $\tau > 0$, thus having a well-defined gradient. While z samples are differentiable, our annotation process requires a discrete selection. Hence, we adopt the Straight-Through (ST) Gumbel-Softmax, which discretizes z through argmax in the forward pass and utilizes the continuous approximation in the backward pass. using the ST Gumbel-Softmax on $[\mathbf{l}_1, \ldots, \mathbf{l}_{|D|}]$, we derive a one-hot matrix $\mathbf{M}_{gumbel} \in \mathbb{R}^{|D|\times 2}$, where each utterance is assigned the vector $[0.0, 1.0]$ if relevant and $[1.0, 0.0]$ otherwise. A vectorial representation $\mathbf{v}_{gumbel} \in \mathbb{R}^{|D|}$ is $\mathbf{M}_{gumbel}(:, 2)$.

Annotation gate. To preserve the gradient flow after classification, it is necessary to identify a differentiable strategy also for dialogue augmentation. Compared to joint learning, we (i) reframe the input as $\mathbf{x_{E2E}}$ by positioning the `<hl></hl>` tokens at the start and end of each utterance, (ii) delineate a gating mechanism to dynamically turn off the wrapping annotation of unselected utterances. If a dialogue line is deemed irrelevant, we replace the embeddings of its `<hl></hl>` tokens with those of space tokens (i.e., " "), mimicking their absence in the input. Mechanically, we obtain the representation of each token $\mathbf{M}_{emb} \in \mathbb{R}^{|\mathbf{x_{E2E}}|\times d}$ by feeding $\mathbf{x_{E2E}}$ to the summarizer's input embedding layer. We construct a mask $\mathbf{v}_{gate} \in \mathbb{R}^{|\mathbf{x_{E2E}}|}$ to cancel undesired special tags, where the binary weight of each token $t_i \in \mathbf{x_{E2E}}$ is established with the following function:

$$\text{mask}(t_i) = \begin{cases} \mathbf{v}_{gumbel}[\text{utterance}(t_i)] & t_i \in \{\langle \texttt{hl}\rangle, \langle \texttt{/hl}\rangle\} \\ 1 & t_i \notin \{\langle \texttt{hl}\rangle, \langle \texttt{/hl}\rangle\}. \end{cases} \tag{7}$$

We build a matrix $\mathbf{M}_{space} \in \mathbb{R}^{|\mathbf{x_{E2E}}|\times d}$ containing space-token embeddings for annotation tags of irrelevant utterances, 0-embeddings otherwise. The matrix supplied to the summarizer is $\left(\mathbf{v}_{gate}^T \odot \mathbf{M}_{emb}\right) + \mathbf{M}_{space}$, while the final loss is $\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{nll}$ as in Section 4.1.

5. Experimental setup

Implementation details, computing infrastructure, and experiment hyperparameters are described in Appendix A. For all the models reported in Section 5.2, we adopt the optimal configurations indicated by the authors, tailored to our hardware capacity.

5.1. Dataset

On the steps of previous research, we carry out experiments on SAMSum [3] and DialogSum [49], two modern testbeds for abstractive dialogue summarization in the English language. SAMSum is prepared by Samsung R&D Institute Poland. Linguists created and wrote conversations to reflect the complexity and proportion of daily topics. DialogSum is gathered from a practice website and three publicly available dialogue datasets, namely Dailydialog [50], DREAM [51], MuTual [52]. For both of these two benchmarks, the style and register are diversified (informal, semi-formal, formal), encompassing a broad spectrum of everyday subjects, such as education, employment, healthcare, chit-chats, meeting organization, shopping, recreation, and travel. Summaries are presented in the third person and offer succinct overviews of the discussed topics. Datasets statistics are detailed in Table 1.

5.2. Models

Classification and summarization modules. Our annotate-then-summarize framework is agnostic to the underlying models. We investigate several transformer-based PLMs with different capacities (i.e., architectures, pretraining schemes), aiming to reuse their linguistic and semantic knowledge. We also fine-tune large language models (LLMs) with > 1B parameters. As for the classifier, we test representation and generative networks to encode each utterance's last token (SEP). Specifically, we put into play RoBERTa [54], DeBERTaV2 [55], GPT-2 [56], and OPT [57]. For Top-p-semantic-overlap, semantic similarity is calculated with a RoBERTa-large model fine-tuned on natural language inference (NLI).[3] As for the summarizer, we evaluate BART [16]—the most popular generative model for dialogue-oriented tasks [4], rooted in denoising pretraining objectives, and FLAN-T5 [58]—a large-scale model fine-tuned with instructions on a mixture of text-to-text tasks (dialogue summarization excluded).

Baselines. We head-to-head compare our DearWatson models with a plethora of competitive abstractive summarization baselines.

- BART, FLAN-T5. Vanilla models fine-tuned on non-augmented dialogues, i.e., without our learnable annotation.
- MV-BART [9]. BART model with multi-view decoder attention layers to incorporate conversation topics and progression stages.
- Coref-ATTN [33]. BART model with an additional coreference-guided attention layer between the encoder and the decoder.
- S-BART [59]. BART model with a multi-granularity decoder incorporating utterance dependency graphs and action graphs.
- BART-DialoGPTAnn [10]. BART model trained on documents augmented with unsupervised topic, keyword, and redundancy annotations from DialoGPT [23].
- SWING [60]. BART model trained with NLI signals to encourage coverage and factuality.
- DialSent [61]. BART model post-trained to rephrase from dialogues to narratives before fine-tuning for dialogue summarization.

The large version is harnessed across each BART-derived model to promote fairness.

5.3. Metrics

We quantify classification effectiveness with recall, precision, F1-score, and accuracy for positive utterance classification T. Following the trail of common practice, we evaluate summarization performance in terms of ROUGE-1/2/L F1 scores [62]. Inspired by Moro et al. [63],

[3] https://huggingface.co/sentence-transformers/roberta-large-nli-stsb-mean-tokens.

Table 1
SAMSum and DialogSum dataset statistics. Word counts are determined by the NLTK tokenizer [53].

# Dialogues		# Participants			# Turns			# Dialogue words			# Summary words		
		Mean	Std	Range	Mean	Std	Range	Mean	Std	Range	Mean	Std	Range
SamSum													
Train	12,460	2.40	0.83	[1, 14]	11.17	6.45	[1, 46]	124.5	94.2	[13, 1017]	23.44	12.72	[2, 73]
Dev	818	2.39	0.84	[2, 12]	10.83	6.37	[3, 30]	121.6	94.6	[18, 691]	23.42	12.71	[4, 68]
Test	819	2.36	0.83	[2, 11]	11.25	6.35	[3, 30]	126.7	95.7	[17, 669]	23.12	12.20	[4, 71]
DialogSum													
Train	12,460	2.01	0.13	[2, 7]	9.49	4.16	[2, 61]	197	97.83	[52, 1389]	31.02	13.51	[7, 212]
Dev	500	2.01	0.13	[2, 4]	9.38	3.99	[2, 29]	194.38	90.81	[55, 672]	28.96	12.63	[7, 89]
Test	500	2.01	0.09	[2, 3]	9.71	4.99	[2, 65]	202.22	108.84	[54, 1258]	25.59	11.01	[6, 96]

Table 2
Relevant utterance classification results on SAMSum and DialogSum test sets (3-runs average). All scores correspond to the annotation strategy that has been empirically identified as the most effective for each dataset, specifically, One-word-overlap for SAMSum and Top-p-word-overlap for DialogSum. Each dataset's results are categorized into two sections. Top: classification-only fine-tuning; bold and underline denote best and second-best scores. Bottom: annotate-then-summarize fine-tuning; ↑ highlights relative improvements.

	Recall	Precision	F1	Accuracy
SAMSum				
GPT-2-XL	<u>86.11</u>	75.97	80.72	80.43
OPT-1.3B	84.10	<u>77.78</u>	<u>80.82</u>	<u>81.00</u>
DeBERTaV2-xlarge	**87.34**	74.27	80.27	79.60
RoBERTa-large	82.15	**79.90**	81.01	81.70
RoBERTa-large(DW_{JL})	83.03 ↑	78.68	80.79	81.24
RoBERTa-large(DW_{E2E})	85.00 ↑	78.43	**81.59** ↑	**81.77** ↑
DialogSum				
GPT-2-XL	<u>67.67</u>	63.35	65.44	70.17
OPT-1.3B	67.16	65.02	66.07	71.22
DeBERTaV2-xlarge	65.57	62.18	63.83	69.07
RoBERTa-large	**69.33**	**68.95**	**69.14**	**74.24**
RoBERTa-large(DW_{JL})	61.38	66.67	63.91	73.30
RoBERTa-large(DW_{E2E})	67.47	<u>67.23</u>	<u>67.35</u>	<u>73.95</u>

we also measure an aggregated ROUGE judgment: $\mathcal{R} = {}^{\text{avg}(r_1,r_2,r_L)}/_{1+\sigma_r^2}$, where $r_{1/2/L} \in [0,1]$ and σ_r^2 is the F1 variance.[4] To refine summary quality assessment and go beyond lexical superficiality, we make use of BERTScore [64] and BARTScore [65], two recently-developed model-based metrics that have been shown to correlate well with human judgments (i.e., semantic coverage, coherence, informativeness, relevance, fluency, and factual consistency dimensions). Higher scores indicate better overall results. Metric settings and interpretability hints are documented in Appendix A.2.

6. Results

6.1. Quantitative evaluation

Utterance classification. Table 2 displays preliminary fine-tuning results about the influence of the PLM choice on the classification component alone. The reported results are based on the most effective annotation strategy identified for each dataset in the downstream summarization task. A thorough exploration of the strategy's impact is detailed in the next paragraph. Notably, RoBERTa stands out from the other models, exhibiting the highest F1 and accuracy scores, despite having up to 4.5x fewer parameters. Accordingly, we opt for RoBERTa as the classification backbone in all our complete solutions, denoted by DW_{JL} (joint learning) and DW_{EE} (end-to-end).

[4] $\mathcal{R}$ penalizes model results with discrepant unigram, bigram, and longest common subsequence overlaps.

Annotation strategy. The way "relevance" is defined for source utterances can have a profound impact on the quality of the generated summary. Table 3 highlights how the choice of different annotation strategies directly affects summarization metrics. Interestingly, the optimal strategy appears to be highly dependent on the dataset at hand, revealing contrasting preferences between SAMSum and DialogSum. To be specific, One-word-overlap appears to be the best choice for SAMSum and the worst for DialogSum. Given the similar nature of the two benchmarks, we posit that a pivotal factor influencing the ultimate performance lies in achieving a harmonious balance between annotated and non-annotated oracle utterances, essentially the equilibrium between positive and negative classification examples. To go into this interplay, we present the percentage of annotated utterances for each strategy-dataset combination. For the top-p heuristics, the annotation percentage is equal to p; we report the best value registered after a grid search in the hyperparameter space (see Table A.7 for details). Adding weight to our hypothesis is the optimal value of p, which closely approaches 42%. Simultaneously, in SAMSum and DialogSum, One-word-overlap annotates approximately ≈48% and ≈64% of the source utterances, respectively, indicating a greater likelihood of false positive relevance labels in the latter. Regarding the evaluation of Top-p-word-overlap and Top-p-semantic-overlap in DialogSum, our findings show that the former excels in joint learning and secures higher performance rankings in end-to-end training. As a result, we opt for Top-p-word-overlap.

Dialogue summarization. Overall results are delighted in Table 4, considering the optimal annotation strategy identified for each dataset. The remarkable outcomes produced by the oracle utterance annotations (up to +5.60/3.71/4.34 ROUGE-1/2/L F1, +4.68 BERTScore F1, +0.132 BARTScore F1) clearly affirm the powerful effect of `<hl></hl>` tagging and set out a promising theoretical upper bound (i.e., perfect classifier with 0 false positives and negatives). Expressly, the annotation confers more benefits as the number of parameters decreases, reaching a peak with BART. When evaluated on SAMSum, our DearWatson models outmatch vanilla summarizers in almost all dominant metrics, especially precision measures. Moving to DialogSum, a BART-based backbone becomes sufficient to get competitive or superior ROUGE F1 scores, while greatly improving semantic and factual metrics (up to +13.82 BERTScore F1 and +0.871 BARTScore F1 compared to SWING and DialSent). In the context of ROUGE metrics, making a definitive choice between DW_{JL} and DW_{EE} proves challenging, as the two training modes exhibit distinct performance gaps in SAMSum and DialogSum. DW_{JL} is more robust and predictable, attributed to its finer error containment, which becomes increasingly challenging as the architecture grows. On the other hand, DW_{EE} consistently favor semantic metrics. In addition, DearWatson multi-task learning positively affects the prediction of relevant utterances, further enhancing RoBERTa capabilities; this becomes particularly evident when promoting more interaction through DW_{EE}. Our FLAN-T5(DW_{JL}) eclipses MV-BART on SAMSum and clearly sets a new state-of-the-art, beating multi-view, coreference-centered, multi-modal, and NLI-based alternatives.

Table 3
Quantitative summarization results on the SAMSum and DialogSum test sets after fine-tuning with different target annotation strategies. Oracle-annotated utterances are specified on the right. Bold and underline denote the best and second-best scores for each dataset.

	ROUGE-F	BERTScore-F	BARTScore-F	Ann. utterances
SAMSum				
BART(DW_{JL}) · One-word-overlap	**44.50**	53.61	−2.784	47.59%
BART(DW_{E2E}) · One-word-overlap	44.33	**53.72**	−2.783	
BART(DW_{JL}) · Top-p-word-overlap	43.91	53.31	−2.787	42.25%
BART(DW_{E2E}) · Top-p-word-overlap	43.52	53.07	**−2.778**	40.7%
BART(DW_{JL}) · Top-p-semantic-overlap	43.79	53.29	−2.799	42.25%
BART(DW_{E2E}) · Top-p-semantic-overlap	43.86	53.06	−2.798	40.7%
DialogSum				
BART(DW_{JL}) · One-word-overlap	37.36	51.19	−2.841	63.67%
BART(DW_{E2E}) · One-word-overlap	37.33	51.40	−2.842	
BART(DW_{JL}) · Top-p-word-overlap	38.29	52.45	−2.847	41.71%
BART(DW_{E2E}) · Top-p-word-overlap	38.60	**52.83**	−2.826	39.91%
BART(DW_{JL}) · Top-p-semantic-overlap	37.59	51.76	**−2.822**	41.71%
BART(DW_{E2E}) · Top-p-semantic-overlap	**38.66**	52.78	−2.834	39.91%

Table 4
Quantitative summarization results on the SAMSum and DialogSum test sets after fine-tuning (3-runs average). Top: abstractive baselines. Bottom: backbone vanilla models and our annotate-then-summarize variants, including oracle utterance annotations, joint learning, and end-to-end learning. Bold and underline denote the best and second-best scores, excluding target-aware oracle results. The green gradient spotlights our relative percentage improvement compared to backbone models (the deeper, the more).

Model	ROUGE-1			ROUGE-2			ROUGE-L			BERTScore			BARTScore		
	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
SAMSum															
MV-BART	57.51	55.85	54.05	30.74	29.49	28.56	53.16	51.54	50.57	53.90	53.46	53.64	−2.915	−2.661	−2.788
Coref-ATTN	56.61	57.12	53.93	29.79	30.68	28.58	52.26	52.49	50.39	53.32	53.89	53.56	−2.915	−2.674	−2.794
S-BART	51.22	56.00	50.70	25.84	28.12	25.50	48.17	51.87	48.08	48.86	52.36	50.57	−3.110	−2.895	−3.003
BART-DialoGPTAnn	54.90	57.90	53.70	29.53	31.22	28.79	51.54	53.72	50.81	50.48	42.95	46.66	−2.975	−3.617	−3.296
SWING	57.19	55.27	53.04	30.34	29.99	28.36	52.91	51.59	50.08	53.75	52.49	53.07	−2.906	−2.728	−2.817
DialSent	55.68	57.26	53.54	30.05	31.14	28.91	51.55	52.82	50.21	52.86	53.92	53.34	−2.95	−2.722	−2.836
BART	58.08	53.93	53.06	30.66	28.74	28.08	53.02	49.83	49.44	54.06	51.52	52.74	−2.901	−2.726	−2.813
Ours · BART(DW_{oracle})	60.73	60.34	58.64	32.70	32.86	31.79	55.04	54.97	53.78	57.44	57.41	57.42	−2.788	−2.574	−2.681
Ours · BART(DW_{JL})*	57.76	55.79	53.92	31.22	30.41	29.16	53.16	51.58	50.43	54.27	53.04	53.61	−2.885	−2.684	−2.784
Ours · BART(DW_{E2E})*	57.80	55.49	53.75	30.80	30.12	28.84	53.23	51.47	50.40	54.39	53.14	53.72	−2.891	−2.675	−2.783
FLAN-T5	**58.36**	56.59	54.38	**32.36**	31.46	30.05	**54.15**	52.79	51.35	**55.10**	53.73	54.36	**−2.822**	−2.668	−2.745
Ours · FLAN-T5(DW_{oracle})	59.45	58.12	56.22	33.15	32.55	31.31	55.33	54.32	53.09	56.69	55.65	56.13	−2.788	−2.612	−2.700
Ours · FLAN-T5(DW_{JL})*	57.90	**58.09**	**55.05**	32.26	**32.56**	**30.63**	53.89	**54.08**	**51.98**	**55.10**	**54.82**	**54.91**	−2.827	**−2.646**	**−2.737**
Ours · FLAN-T5(DW_{E2E})	57.94	56.57	54.13	32.14	31.33	29.86	53.99	52.87	51.28	54.74	53.37	54.00	−2.828	−2.659	−2.744
DialogSum															
SWING	**54.47**	44.85	47.67	**24.99**	20.82	**21.99**	**50.79**	43.28	45.72	45.81	41.04	43.45	−3.576	−3.343	−3.459
DialSent	46.46	**52.16**	47.60	21.05	**24.34**	21.76	44.60	**49.26**	**45.79**	36.61	41.36	39.01	−3.760	−3.633	−3.697
BART	53.37	43.83	46.55	23.87	20.00	21.00	49.82	42.32	44.70	54.59	48.33	51.45	−2.865	−2.824	−2.844
Ours · BART(DW_{oracle})	58.67	48.01	51.24	27.50	22.95	24.27	53.60	45.38	48.13	57.61	51.09	54.34	−2.741	−2.730	−2.735
Ours · BART(DW_{JL})*	53.78	45.47	47.79	24.05	20.84	21.62	49.93	43.45	45.47	**55.23**	43.45	45.47	**−2.861**	−2.833	−2.847
Ours · BART(DW_{E2E})*	52.53	47.38	**48.18**	23.70	21.79	21.95	48.84	44.82	45.66	54.67	**50.97**	**52.83**	−2.867	**−2.785**	**−2.826**

* Statistical significance (Pitman's permutation test, $p < 0.05$).

Annotation statistics. Table 5 wraps up fine-grained and coarse-grained annotation coverage statistics in augmented dialogue. On average, models trained in a joint learning mode are more likely to label short utterances.

Impact of speakers and utterances. Fig. 3 breaks down the $\mathcal{R}$ effectiveness by number of participants and utterances. Please note that DialogSum only has 2 or 3 speakers (see Table 1). Irrespective of the model or learning approach, as the number of utterances increases, ROUGE scores gradually decline in SAMSum. In contrast, in DialogSum, they exhibit a less predictable pattern, characterized by alternating spikes of good and poor performance. Importantly, SAMSum dialogues with 5+ speakers are accompanied by a marked $\mathcal{R}$ reduction. We exclude speaker number classes with fewer than three occurrences.

6.2. Cross-attention analysis and interpretability

We analyze the cross-attention values between the input dialogue and the generated summary to explicate the performance boost permitted by relevant utterance annotations. Fig. 4 contrasts the behavior of BART and BART(DW_{JL}) on a qualitative SAMSum example. The standard BART model places little focus on salient dialogue tokens such as `[bring, piece, later, on]`, causing essential facts to be missing from the resume. In BART(DW_{JL}), we observe a discernible emphasis on the importance of `<hl></hl>` tokens. Moreover, the model concentrates more on the pertinent facts while attenuating attention toward others. During inference, selected utterances act as evidence for the final generation, making the model more interpretable. Fig. 5 portrays input–output qualitative case studies that elucidate the advantages of our models.

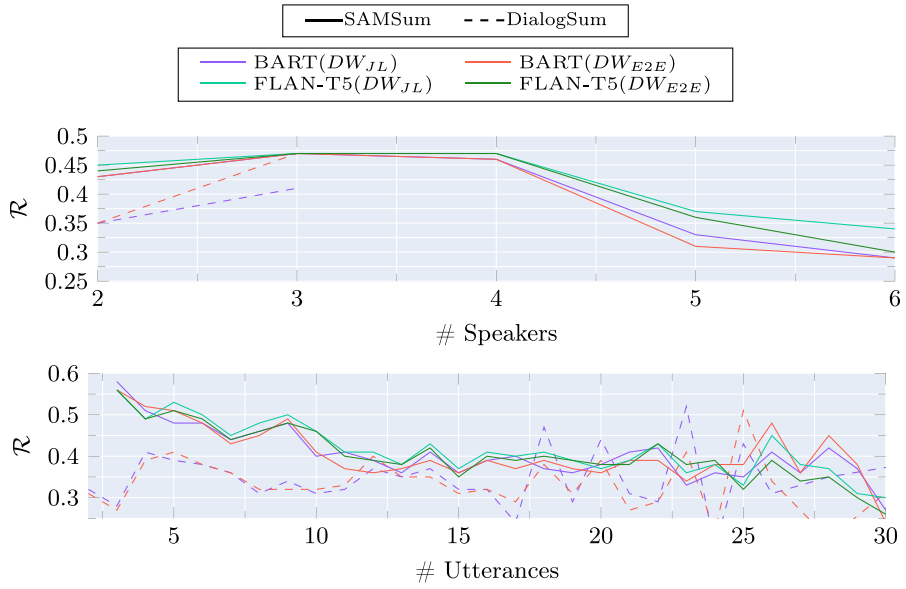


Fig. 3. Relation between overall summarization ROUGE scores ($\mathcal{R}$) and the number of speakers/utterances in conversations.

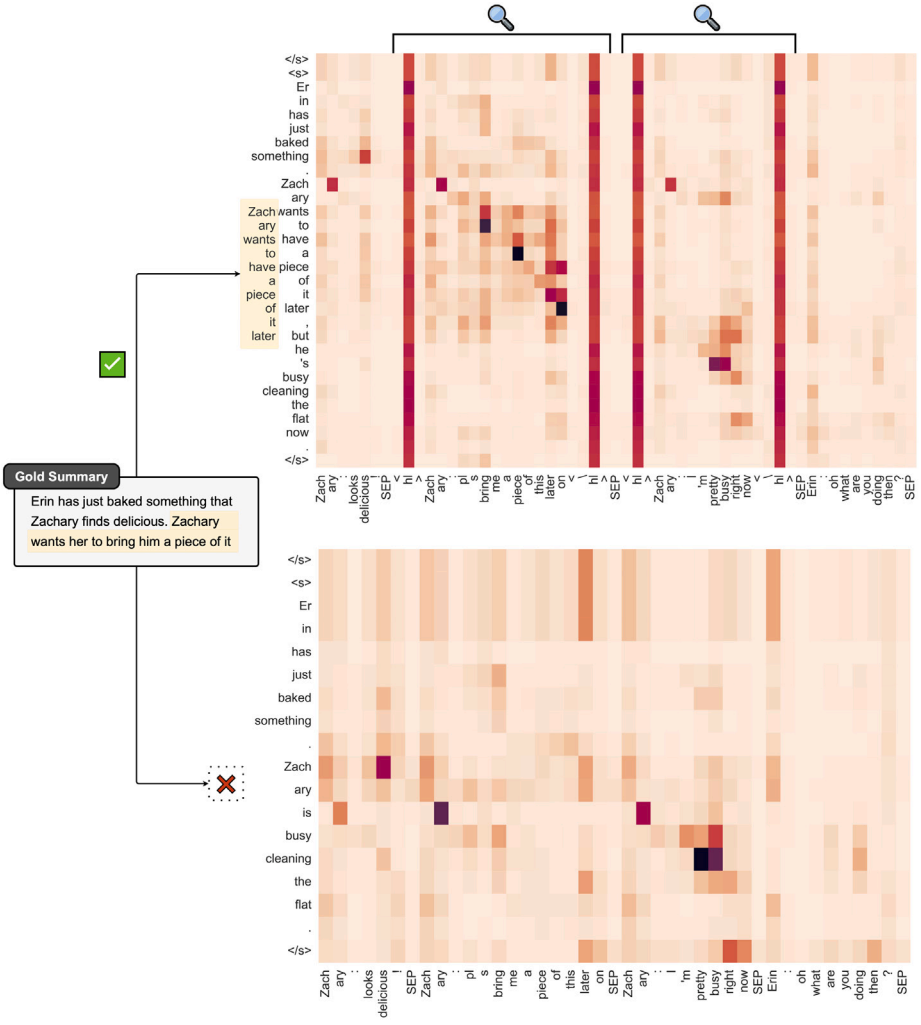


Fig. 4. Attention matrix visualization, the more intense the color, the higher the weight. A comparison between BART(DW_{JL}) (left) and BART (right). The vertical axis is the generated summary, and the horizontal axis is the source dialogue. Presented tokens are partial for ease of perception. Our framework guides the model to emphasize tokens that are more likely to be relevant for the summary.

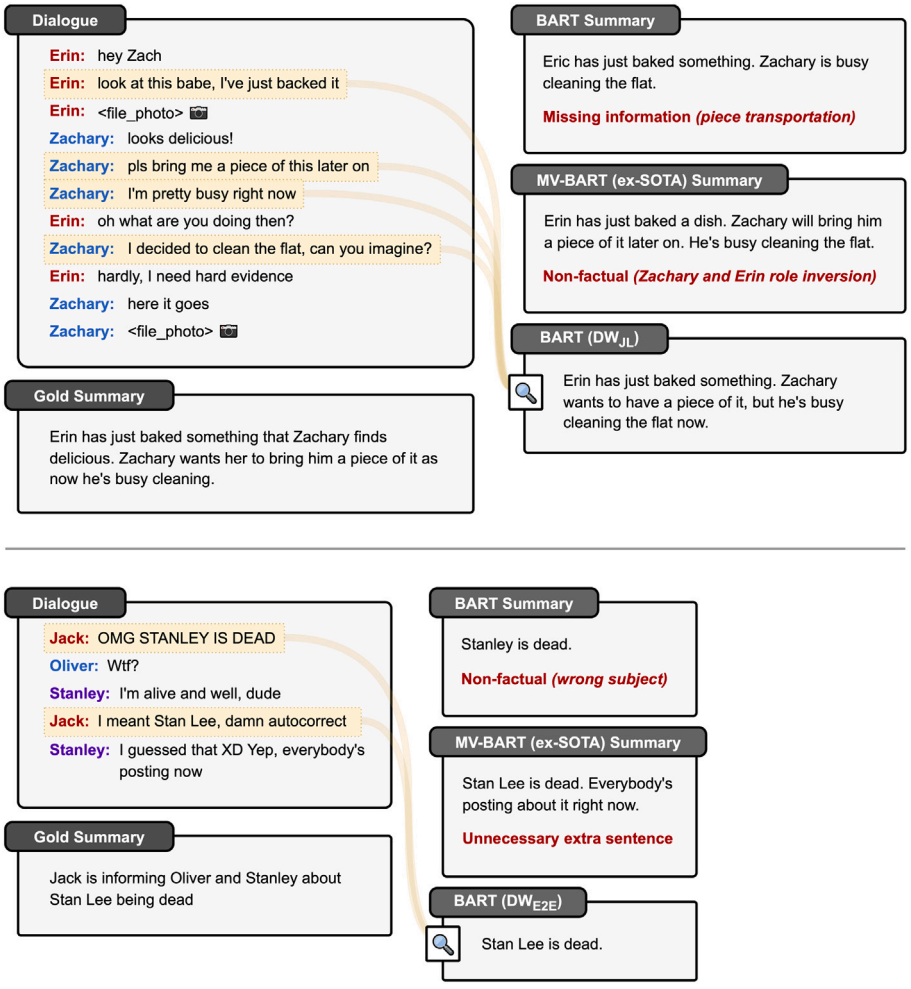


Fig. 5. Qualitative examples of predicted relevant utterances (highlighted in yellow) and their assistance to high-quality summarization and interpretability. Red text indicates generation errors. Taking BART as the backbone, we illustrate the inference of our fine-tuned joint learning model (left) and end-to-end model (right).

Table 5
Average relevant utterance annotation and generation statistics. % of annotated utterances out of the total. % of annotated words out of the total. Number of generated words. Number of generated tokens.

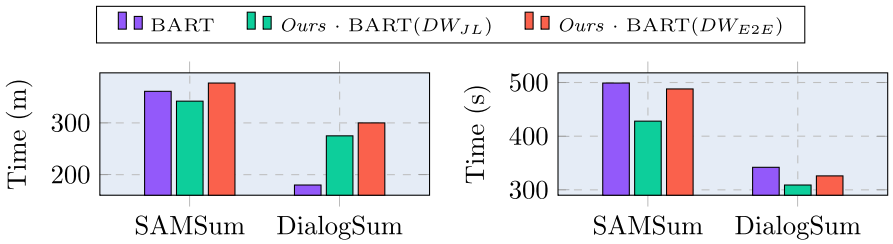
Dataset	Model	Ann. utterances	Ann. words	Gen. words	Gen. tokens
SAMSum	BART	50.14	65.3	25.7	28.62
	BART(DW_{JL})	51.49	67.05	24.74	27.66
	BART(DW_{E2E})	51.49	67.05	25.0	27.95
	FLAN-T5	52.58	67.97	24.97	30.94
	FLAN-T5(DW_{JL})	52.58	67.97	23.92	29.59
	FLAN-T5(DW_{E2E})	49.92	64.85	24.79	30.76
DialogSum	BART	35.47	43.88	30.91	34.66
	BART(DW_{JL})	51.49	67.05	29.66	33.48
	BART(DW_{E2E})	39.97	47.99	27.99	31.68

6.3. Efficiency

Efficiency takes the spotlight in Fig. 6, proving the practical usability of the proposed method. In contrast to plain models lacking an annotator, the training process for our DW_{JL} and DW_{E2E} classifier+summarizer models demands an additional 2 h at most on DialogSum, while it only takes up to 16 extra minutes on SAMSum. During inference, RoBERTa-large runs in ≈8 s for the entire DialogSum test set and ≈13 s for the SAMSum test set. As a result, the additional cost introduced by the annotator is negligible and cannot exceed the runtime variability effect between different runs. In fact, the overall inference times of DW_{JL} and DW_{E2E} are even shorter than those of BART. The shorter average length of the summaries produced by our models adds to the rationale for this time efficiency (Table 5).

6.4. Human evaluation

In the quest to better gauge summarization merits, we conduct an in-depth human evaluation of three highly-comparable models on SAMSum: the previous state-of-the-art holder MV-BART and ours BART(DW_{JL}), BART(DW_{EE}). The exclusive focus on SAMSum is firmly supported by its widespread popularity and frequent adoption as the sole benchmark in previous research work [9,22,33]. Motivated by [60,66,67], we use a direct comparison strategy, which has been shown to be more reliable, sensitive, and less labor intensive than rating scales. We sample 50 instances from the test set. For each instance, 3 English-proficient graders are presented with summaries inferred from 2 out of 3 sources and asked to select the better one with respect to 3 dimensions. A "Tie" is declared if a judge perceives the two summaries to be of equal quality. When all possible pair combinations are scrutinized, the total number of preference labels per annotator is 450. We randomize the order of pairs and summaries per example to guard the rating against being gamed. Zooming in, the rating axes are defined as follows. *Recall* considers whether the generated summary covers all the target content units. *Precision* checks if the generated summary covers only the target content units (i.e., no superfluous or redundant information). *Faithfulness* examines whether the generated summary is factually consistent with the dialogue. The final score of



(a) Training time. (b) Inference time.

Fig. 6. Runtime analysis (3-runs average) over the SAMSum and DialogSum test sets, BART-based backbone. Our model results include both annotation and generation times.

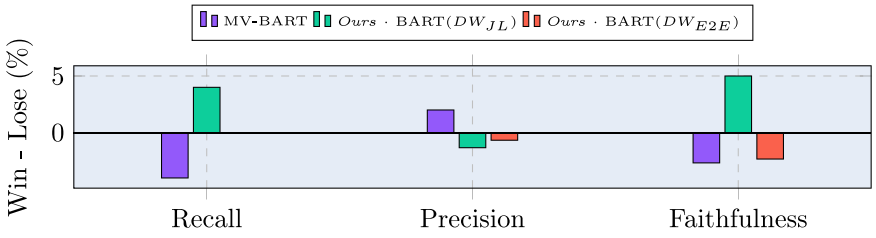


Fig. 7. Human evaluation results. DearWatson joint learning and end-to-end learning achieve significantly higher recall and faithfulness than the previous state-of-the-art (student t-test, p<0.05), corroborating the benefits of supervised input augmentation.

each model is given by the percentage of times that its summaries are selected as the better ones, minus the percentage of times that they are not. Appendix B illustrates our setup with human instructions.

The results are showcased in Fig. 7. Sampled instances and human judges are published for transparency and further applications.[5] The average Kendall coefficient among all inter-annotator agreements is 45.25%. The annotation process took approximately 6 h per judge, 18 h in total. DearWatson models rank better on recall (+200% joint, +100% end-to-end) and faithfulness (+2324% joint, +103% end-to-end) at the cost of a moderate drop in prediction (−83% joint, −67% end-to-end). These findings reflect that the summaries inferred with our framework cover more semantic facts presented in the ground truth, successfully addressing the missing information issue. Conversely, they suggest a moderate correlation between automatic metrics and desired output properties, confirmed in Section 6.6.

6.5. ChatGPT evaluation

Scaling the model or data size has primarily raised the performance bar of NLP tasks, boosting model capacity and showing up emergent abilities [68–70]. Current LLMs are general-purpose language task solvers (to some extent) and are often regarded as an initial form of artificial general intelligence [71]. In response to these achievements, newly published work has provided meta-evaluations of ChatGPT in zero-shot settings, attempting to ascertain whether it can evaluate text like a human expert. Gilardi et al. [72] reveal that ChatGPT outperforms MTurk crowd-workers in several annotation tasks. Wang et al. [73] substantiate that ChatGPT surpasses previous automatic metrics in abstractive summarization and attains state-of-the-art correlation with human judgments, making it a premier NLG evaluator. In this paper, we deploy ChatGPT to rank dialogue summarizers for the first time in the literature. In detail, we consider the same evaluation setting described in Section 6.4 (i.e., sample, models, quality dimensions) and give aspect-specific instructions to prompt the reference-based assessment of the generated summary on a 3-point Likert scale. Fig. 8 documents average results per model and metric. Details are reported in Appendix C.

[5] Annotations will be released in case of acceptance.

6.6. Correlation with human judgment

We check out the correlation between automatic metrics (ChatGPT included) and human judgments in terms of semantic recall, precision, and faithfulness. We first convert human evaluation results and automatic metric scores to a scale of {−1, 0, 1}, corresponding to {LOSE, TIE, WIN}. Fig. 9 depicts the results with Kendall's Tau [74] as the correlation measure, which evaluates the ordinal association between two quantities. Note that Kendall–Tau ranges in $[-1, 1]$, with 1 denoting a perfect positive association. Remarkably, ChatGPT correlates the most with humans, representing the most suitable technique to measure the degree to which a model resolves missing information. On the contrary, BERTScore and BARTScore perform surprisingly poorly. We hypothesize that this is due to their model-based nature and pretraining on documents different from conversations. We find ROUGE-L recall being the best metric for assessing factuality. Despite the higher proficiency of ChatGPT in capturing quality dimensions, its correlation with human raters reaches a maximum of 0.2. This indicates that ChatGPT, although a valuable metric, cannot completely substitute human judgment in the abstractive dialogue summarization field. In general terms, all metrics struggle to measure *semantic* fact recall, deviating noticeably from human labels.

7. Conclusion

In this paper, we introduce DearWatson, a new annotate-then-generate framework for abstractive dialogue summarization. Drawing inspiration from the human distillation process, we train a model to predict relevant utterances and consciously build a resume with respect to the essential facts. To achieve this, we jointly optimize the placement of relevance-marker tokens in the input and the summarization of the augmented dialogue. Experiments and ablation studies on SAMSum and DialogSum demonstrate that our models set new state-of-the-art results, significantly improving semantic recall and faithfulness. Cross-attention analysis unveils that our framework instructs pretrained language models to better select and preserve summary-worth content. At inference time, classified utterances also provide evidence for the output, favoring interpretability. To provide insights for future research, we measure the correlation between reported automatic metrics, prompt-guided ChatGPT scores, and human judgments.

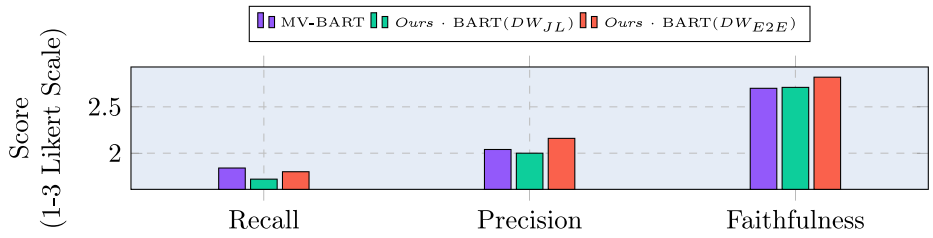


Fig. 8. ChatGPT evaluation results. DearWatson end-to-end models exhibit higher precision and faithfulness.

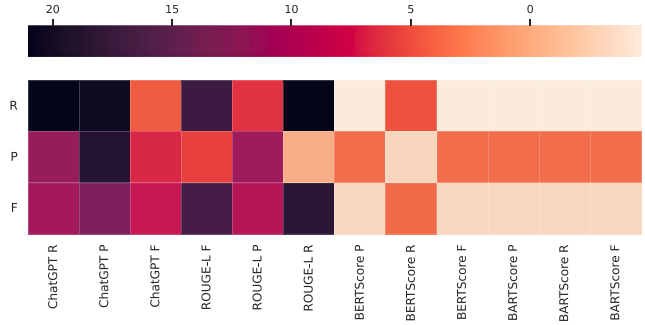


Fig. 9. Kendall–Tau correlation (%) of automatic metrics with human judgments.

CRediT authorship contribution statement

Paolo Italiani: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Giacomo Frisoni:** Conceptualization, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Gianluca Moro:** Conceptualization, Methodology, Investigation, Resources, Writing – review & editing, Supervision, Project administration. **Antonella Carbonaro:** Writing – review & editing, Project administration. **Claudio Sartori:** Writing – review & editing, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The datasets used in this research work are publicly available (CC BY-NC-ND 4.0).

Acknowledgments

This research is partially supported by (i) the Complementary National Plan PNC-I.1 "Research initiatives for innovative technologies and pathways in the health and welfare sector" D.D. 931 of 06/06/2022, DARE—DigitAl lifelong pRevEntion initiative, code PNC0000002, CUP B53C22006450001, (ii) the PNRR—M4C2—Investment 1.3, Extended Partnership PE00000013, FAIR—Future Artificial Intelligence Research, Spoke 8 "Pervasive AI", funded by the European Commission under the NextGeneration EU program. The authors thank the Maggioli Group[6] for partially supporting the Ph.D. scholarship granted to P. Italiani.

[6] https://www.maggioli.com/who-we-are/company-profile.

Appendix A. Reproducibility

A.1. Implementation and training details

Hardware setup and environment. Each experiment is performed on a workstation using a single Nvidia GeForce RTX3090 GPU with 24 GB of dedicated memory, 64 GB of RAM, and an Intel® Core™ i9-10900X1080 CPU @ 3.70 GHz. The reference operating system is Ubuntu 20.04.3 LTS. To elevate portability and consistency, our development environment is constructed on top of a docker container with a de-facto standard HuggingFace image.[7]

Dataset preprocessing. We download SamSum from HuggingFace Datasets.[8] We retain the original text content of conversations, encompassing cased words, slang words, typos, and emoticons. The ":" token is included in the speaker span $\mathbf{p}_i$. We replace newlines with `[SEP]` utterance segmenters.

Models. Table A.6 enumerates the models used in this study, linking to specific versions. They are transformer-based, have a subword vocabulary, and are pre-trained on massive corpora through denoising self-supervised tasks, i.e., reconstruction of artificially masked or corrupted spans. Broadly speaking, self-supervision is a powerful technique to address the challenge of limited labeled data, finding success in various applications such as information retrieval [75] and gene function discovery [76,77]. Given the contained size of SAMSum dialogues, no model has a maximum input size that requires truncation. It is worth mentioning that early experiments were also made with DialogLED [78] as backbone—an efficient model pre-trained for dialogue understanding and summarization on long meeting and TV series transcripts. However, the different nature and domain of pretraining dialogues ended up being unsuitable for the objectives of this paper.

Baselines. For each summarization baseline, we rest on the official SAMSum-specific checkpoints released by the authors, which we employ to re-run inferences and calculate metric scores. The only exception is Coref-ATTN, for which the model has not been released, and we refer directly to the official predictions.

Fine-tuning. Our code is founded on Python 3.10.8, PyTorch 1.12.0 [79], and HuggingFace Transformers [80]. We train each model for five epochs and select the best checkpoints in the validation set with $\mathcal{R}$ score. We choose the AdamW optimizer [81] and leave 42 as the default global seed. Sticking to Jang et al. [43], we anneal the Gumbel softmax value τ according to a high-low schedule. Both BART and FLAN-T5 are trained with teacher forcing: at training time, the inputs are previous tokens from the ground truth; at test time, the inputs are prior tokens predicted by the decoder. To make fine-tuning of GPT-2, OPT, and FLAN-T5 possible with our hardware, we exploit PEFT.[9] Precisely, we carry out 8-bit model quantization and adopt Low-Rank Adaption (LoRA) [82] to only learn a small number of extra model

[7] https://hub.docker.com/r/huggingface/transformers-pytorch-gpu.
[8] https://huggingface.co/datasets/samsum.
[9] https://github.com/huggingface/peft.

Table A.6
List of the classification (top) and summarization (bottom) models used in this study.

Model	# Params	Architecture			URL
		E	D	Details	
DeBERTa-V2-xlarge	900M	✓		24-layers, 1536-hidden, 24-heads	https://huggingface.co/microsoft/deberta-v2-xlarge
RoBERTa-large	355M	✓		24-layers, 1024-hidden, 16-heads	https://huggingface.co/roberta-large
GPT-2-XL	1.5B		✓	48-layers, 1600-hidden, 25-heads	https://huggingface.co/gpt2-xl
OPT-1.3B	1.3B		✓	24-layers, 2048-hidden, 32-heads	https://huggingface.co/facebook/opt-1.3b
BART-large	406M	✓	✓	12-layers, 1024-hidden, 16-heads	https://huggingface.co/facebook/bart-large
FLAN-T5-XXL	11B	✓	✓	24-layers, 4096-hidden, 64-heads	https://huggingface.co/google/flan-t5-xxl
SWING (BART-large)	406M	✓	✓	12-layers, 1024-hidden, 16-heads	https://github.com/amazon-science/AWS-SWING
MV-BART-large	~406M	✓	✓	12-layers, 1024-hidden, 16-heads	https://github.com/SALT-NLP/Multi-View-Seq2Seq
Coref-ATTN (BART-large)	~406M	✓	✓	12-layers, 1024-hidden, 16-heads	hhttps://github.com/seq-to-mind/coref_dial_summ
DialSent (BART-large)	~406M	✓	✓	12-layers, 1024-hidden, 16-heads	https://github.com/jiaqisjtu/dialsent-pgg

Table A.7
Explored hyperparameters along with their empirical search grid. Training time (top) and inference time (bottom).

Hyperparameter	Search space
Top-p annotation[b]	{0.30, 0.40, 0.41, …, 0.45[a](DW_{E2E}), …, 0.47[a](DW_{JL}), …, 0.50}
Dropout rate	0.1
Learning rate	{1e−5, 2e−5[a], 3e−5. 4e−5}, linear scheduler
Optimizer	0.9 β_1, 0.999 β_2, 1e−2 weight decay
Batch size	2
Epochs	5 (validation every epoch)
τ[b]	{1, …, 14[a], …, 21}
Lora_dropout[c]	0.1
Lora_rank[c]	{16[a], 32, 64}
Lora_alpha[c]	{32[a], 64, 128}
Decoding strategy	beam search, n_beams = 5 min_length = 8, max_length = 100 repetition_penalty = 1

[a] The final picked values.
[b] Specific for end-to-end strategies.
[c] Specific for large language models.

parameters. Training our best model, FLAN-T5(DW_{JL}), requires 20 GB VRAM and 25 h; 2.13 kg CO2e carbon footprint, 16.57 kWh energy needed.

Hyperparameters. We list the hyperparameters used to train our DeAR-Watson models in Table A.7. Surveyed values include default settings [16,58]; final choices result from a grid search.

Number of parameters. The architectural sizes of our solutions equal the sum of the classifier and summarizer parameters, independently of the training strategy. By combining a RoBERTa-large annotator and a BART-large generator, our BART($DW_{JL/E2E}$) model is ~761.6M (all trained). Similarly, FLAN-T5($DW_{JL/E2E}$) is ~3.2B (of which only 363.8M are trained).

Experiment tracking. We track all our trainings with Weights & Biases[10] and monitor CO2 emissions with CodeCarbon.[11]

A.2. Metrics

We quantify automatic metric scores using NLG-Metricverse [83]. Note that different libraries may result in different scores. Table A.8 lists all hyperparameters. Owing to the grander correlation with human judgment, we compute BERTScore with DeBERTa-xlarge instead of the default RoBERTa-large, as recommended by the authors from version 0.3.11. To increase interpretability and avoid slight range variations, we set `rescale_with_baseline=True`. Note that BARTScore computes the generation probability $p(\mathbf{y}|\mathbf{x},\theta)$ of a sequence $\mathbf{y}$ conditioned on another sequence $\mathbf{x}$, where θ are the weights of a BART model. Because of this generative approach, the evaluation dimensions vary depending on how $\mathbf{y}$ and $\mathbf{x}$ are defined. As for the other metrics, we consider the Recall, Precision, and F1 settings, thereby feeding BARTScore with a summary hypothesis ($\mathbf{h}$) and a summary reference ($\mathbf{r}$). Recall ($\mathbf{h} \to \mathbf{r}$, $p(\mathbf{r}|\mathbf{h},\theta)$) quantifies how easily a gold reference could be generated by the hypothesis (i.e., semantic coverage). Precision ($\mathbf{r} \to \mathbf{h}$, $p(\mathbf{h}|\mathbf{r},\theta)$) assesses how likely the summary hypothesis could be constructed based on the gold reference. F score ($\mathbf{h} \leftrightarrow \mathbf{r}$) takes the arithmetic average of recall and precision.

[10] https://wandb.ai.
[11] https://github.com/mlco2/codecarbon.

Appendix B. ChatGPT evaluation details

We leverage the ChatGPT API and treat the `gpt-3.5-turbo` model as a reference-based metric for judging artificial summaries. We feed predictions individually, utilizing the prompt in Fig. B.10—which, after several attempts, results as the best one. The prompt makes the model aware of task and aspect details, forcing ChatGPT to output scores only.

- *Recall.* 1 = "reference contents are not covered at all". 3 = "reference contents are fully covered".
- *Precision.* 1 = "the generated summary has many contents not covered by the reference (unnecessary or redundant information)". 3 = "the generated summary contains only reference contents".
- *Faithfulness.* 1 = "the generated summary is not semantically consistent with the dialogue (hallucinations, polarity inversions, entity misusage)". 3 = "the generated summary is perfectly factual with respect to the dialogue".

A new chat session is created for every summary to ensure that ChatGPT results (i.e., single digits) are not influenced by the annotation history. As for decoding hyperparameters, we leave default values (e.g., temperature = 1, top_p = 1).

Table A.8
Hyperparameters initialization for utilized NLG metrics. Arrows indicate the reading key (i.e., ↑ = higher is better).

Metric	Definition	Bound	Hyperparameters
ROUGE	Unigrams, bigrams, and longest common subsequence lexical overlaps	$[0, 1]$ ↑	`rouge_types=[‘‘rouge1’’,‘‘rouge2’’,‘‘rougeL’’], use_aggregator=True, use_stemmer=True, metric_to_select=‘‘fmeasure’’`
BERTScore	IDF-weighted n-gram hard-alignment via contextualized embeddings	$[-1, 1]$* ↑	`model_type=‘‘microsoft/deberta-xlarge-mnli’’, idf=True, batch_size=64, nthreads=4, rescale_with_baseline=True,* use_fast_tokenizer=False, return_average_scores=False`
BARTScore	Semantic multi-perspective evaluation as the logarithmic probability of generating a text conditioned on another one	$]-\infty, 0[$ ↑	`model_checkpoint=‘‘bartscore-large-cnn’’, batch_size=4, segment_scores=False`

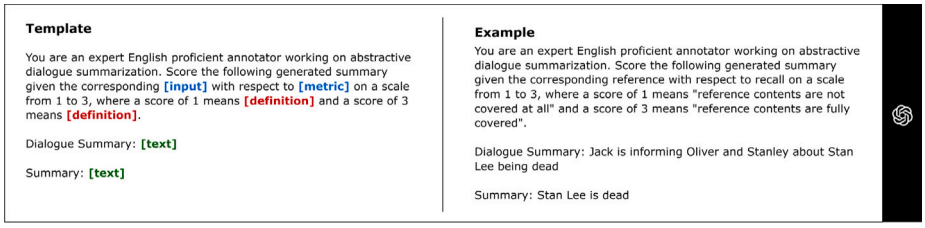


Fig. B.10. ChatGPT annotation prompt (target: value on a 3-point Likert scale). `[metric]` equals “recall”, “precision”, or “faithfulness”. The `[input]` can be a “resume” (recall/precision) or a “document” (faithfulness). `[definition]` spans explain the meaning of boundary min/max scores for the metric under investigation.

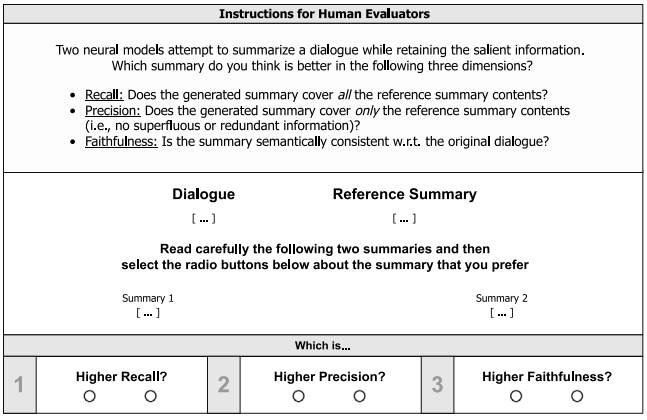


Fig. C.11. Human assessment interface.

Appendix C. Human evaluation details

The interface with human evaluation instructions is sketched in Fig. C.11.

Appendix D. Scientific artifacts

The licenses for all the models and software used in this paper are listed below in parentheses: NLTK (Apache License 2.0), py-ROUGE (Apache License 2.0), BERTScore (MIT License) BARTScore (Apache License 2.0), SAMSum (CC BY-NC-ND 4.0), GPT-2 (MIT License), OPT (non-commercial), DeBERTa (MIT License), RoBERTa (GPL-2.0 License), MV-BART (MIT License), Coref-ATTN (not specified), S-BART (MIT License), BART-DialoGPTAnn (not specified), SWING (Apache License 2.0), BART (MIT License), FLAN-T5 (Apache 2.0), DialSent (not specified).

References

[1] Statista, Most popular global mobile messaging apps 2022, 2022, https://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/. Online; accessed 12 April 2023.
[2] H. Sacks, E.A. Schegloff, G. Jefferson, A simplest systematics for the organization of turn taking for conversation, in: Studies in the Organization of Conversational Interaction, Elsevier, 1978, pp. 7–55, http://dx.doi.org/10.1016/B978-0-12-623550-0.50008-2.
[3] B. Gliwa, I. Mochol, M. Biesek, A. Wawer, SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization, in: Proceedings of the 2nd Workshop on New Frontiers in Summarization, ACL, Hong Kong, China, 2019, pp. 70–79, http://dx.doi.org/10.18653/v1/D19-5409, URL https://aclanthology.org/D19-5409.
[4] X. Feng, X. Feng, B. Qin, A survey on dialogue summarization: Recent advances and new frontiers, in: IJCAI, ijcai.org, 2022, pp. 5453–5460.
[5] M. Li, L. Zhang, H. Ji, R.J. Radke, Keep meeting summaries on topic: Abstractive multi-modal meeting summarization, in: ACL, ACL, Florence, Italy, 2019, pp. 2190–2196, http://dx.doi.org/10.18653/v1/P19-1210, URL https://aclanthology.org/P19-1210.
[6] K. Zechner, Automatic summarization of open-domain multiparty dialogues in diverse genres, Comput. Linguist. 28 (4) (2002) 447–485.
[7] G. Murray, S. Renals, J. Carletta, Extractive summarization of meeting recordings, in: INTERSPEECH, ISCA, 2005, pp. 593–596.
[8] Z. Liu, A. Ng, S.L.S. Guang, A.T. Aw, et al., Topic-aware pointer-generator networks for summarizing spoken conversations, in: ASRU, IEEE, 2019, pp. 814–821.
[9] J. Chen, D. Yang, Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization, in: EMNLP, ACL, Online, 2020, pp. 4106–4118, http://dx.doi.org/10.18653/v1/2020.emnlp-main.336, URL https://aclanthology.org/2020.emnlp-main.336.
[10] X. Feng, X. Feng, L. Qin, B. Qin, et al., Language model as an annotator: Exploring dialoGPT for dialogue summarization, in: ACL-IJCNLP, ACL, Online, 2021, pp. 1479–1491, http://dx.doi.org/10.18653/v1/2021.acl-long.117, URL https://aclanthology.org/2021.acl-long.117.
[11] V. Srivastava, S. Bhat, N. Pedanekar, A few good sentences: Content selection for abstractive text summarization, in: D. Koutra, C. Plant, M.G. Rodriguez, E. Baralis, F. Bonchi (Eds.), Machine Learning and Knowledge Discovery in Databases: Research Track - European Conference, ECML PKDD 2023, Turin, Italy, September 18-22, 2023, Proceedings, Part IV, in: Lecture Notes in Computer Science, vol. 14172, Springer, 2023, pp. 124–141, http://dx.doi.org/10.1007/978-3-031-43421-1_8.
[12] E. Jang, S. Gu, B. Poole, Categorical reparameterization with gumbel-softmax, in: ICLR (Poster), OpenReview.net, 2017.
[13] Y. Chen, Y. Liu, L. Chen, Y. Zhang, DialogSum: A real-life scenario dialogue summarization dataset, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp.

5062–5074, http://dx.doi.org/10.18653/v1/2021.findings-acl.449, URL https://aclanthology.org/2021.findings-acl.449.

[14] I.A.M. Huijben, W. Kool, M.B. Paulus, R.J.G. van Sloun, A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning, IEEE Trans. Pattern Anal. Mach. Intell. 45 (2021) 1353–1371, URL https://api.semanticscholar.org/CorpusID:238259238.

[15] J. Zhang, Y. Zhao, M. Saleh, P.J. Liu, PEGASUS: pre-training with extracted gap-sentences for abstractive summarization, in: ICML, in: PMLR, vol. 119, PMLR, 2020, pp. 11328–11339.

[16] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, et al., BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, in: ACL, ACL, 2020, pp. 7871–7880.

[17] G. Moro, L. Ragazzi, L. Valgimigli, D. Freddi, Discriminative marginalized probabilistic neural method for multi-document summarization of medical literature, in: ACL (1), ACL, 2022, pp. 180–189.

[18] A. Ghadimi, H. Beigy, Hybrid multi-document summarization using pre-trained language models, Expert Syst. Appl. 192 (2022) 116292, http://dx.doi.org/10.1016/J.ESWA.2021.116292.

[19] Y. Zou, B. Zhu, X. Hu, T. Gui, Q. Zhang, Low-resource dialogue summarization with domain-agnostic multi-source pretraining, in: M. Moens, X. Huang, L. Specia, S.W. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, Association for Computational Linguistics, 2021, pp. 80–91, http://dx.doi.org/10.18653/V1/2021.EMNLP-MAIN.7.

[20] G. Moro, L. Ragazzi, Semantic self-segmentation for abstractive summarization of long documents in low-resource regimes, in: AAAI, AAAI Press, 2022, pp. 11085–11093.

[21] G. Moro, L. Ragazzi, L. Valgimigli, G. Frisoni, C. Sartori, G. Marfia, Efficient memory-enhanced transformer for long-document summarization in low-resource regimes, Sensors 23 (7) (2023) http://dx.doi.org/10.3390/s23073542, URL https://www.mdpi.com/1424-8220/23/7/3542.

[22] Z. Liu, N. Chen, Controllable neural dialogue summarization with personal named entity planning, in: ACL, ACL, Online and Punta Cana, Dominican Republic, 2021, pp. 92–106, http://dx.doi.org/10.18653/v1/2021.emnlp-main.8, URL https://aclanthology.org/2021.emnlp-main.8.

[23] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, et al., DIALOGPT : Large-scale generative pre-training for conversational response generation, in: ACL, ACL, Online, 2020, pp. 270–278, http://dx.doi.org/10.18653/v1/2020.acl-demos.30, URL https://aclanthology.org/2020.acl-demos.30.

[24] S. Bao, H. He, F. Wang, H. Wu, et al., PLATO: Pre-trained dialogue generation model with discrete latent variable, in: ACL, ACL, Online, 2020, pp. 85–96, http://dx.doi.org/10.18653/v1/2020.acl-main.9, URL https://aclanthology.org/2020.acl-main.9.

[25] C.-S. Wu, S.C. Hoi, R. Socher, C. Xiong, TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue, in: EMNLP, ACL, Online, 2020, pp. 917–929, http://dx.doi.org/10.18653/v1/2020.emnlp-main.66, URL https://aclanthology.org/2020.emnlp-main.66.

[26] Y. Cao, W. Bi, M. Fang, D. Tao, Pretrained language models for dialogue generation with multiple input sources, in: EMNLP, ACL, Online, 2020, pp. 909–917, http://dx.doi.org/10.18653/v1/2020.findings-emnlp.81, URL https://aclanthology.org/2020.findings-emnlp.81.

[27] X. Gao, Y. Zhang, M. Galley, C. Brockett, et al., Dialogue response ranking training with large-scale human feedback data, in: EMNLP, ACL, Online, 2020, pp. 386–395, http://dx.doi.org/10.18653/v1/2020.emnlp-main.28, URL https://aclanthology.org/2020.emnlp-main.28.

[28] J.-C. Gu, C. Tao, Z. Ling, C. Xu, et al., MPC-BERT: A pre-trained language model for multi-party conversation understanding, in: ACL, ACL, Online, 2021, pp. 3682–3692, http://dx.doi.org/10.18653/v1/2021.acl-long.285, URL https://aclanthology.org/2021.acl-long.285.

[29] M. Zhong, Y. Liu, Y. Xu, C. Zhu, et al., DialogLM: Pre-trained model for long dialogue understanding and summarization, in: AAAI, AAAI Press, 2022, pp. 11765–11773.

[30] G. Domeniconi, G. Moro, A. Pagliarani, R. Pasolini, Markov chain based method for in-domain and cross-domain sentiment classification, in: A.L.N. Fred, J.L.G. Dietz, D. Aveiro, K. Liu, J. Filipe (Eds.), KDIR 2015 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, part of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2015), Volume 1, Lisbon, Portugal, November 12-14, 2015, SciTePress, 2015, pp. 127–137, http://dx.doi.org/10.5220/0005636001270137.

[31] G. Frisoni, G. Moro, Phenomena explanation from text: unsupervised learning of interpretable and statistically significant knowledge, in: S. Hammoudi, C. Quix, J. Bernardino (Eds.), Data Management Technologies and Applications - 9th International Conference, DATA 2020, Virtual Event, July 7-9, 2020, Revised Selected Papers, in: Communications in Computer and Information Science, 1446, Springer, 2020, pp. 293–318, http://dx.doi.org/10.1007/978-3-030-83014-4_14.

[32] G. Frisoni, P. Italiani, S. Salvatori, G. Moro, Cogito ergo summ: Abstractive summarization of biomedical papers via semantic parsing graphs and consistency rewards, in: AAAI, AAAI Press, 2023, pp. 1–9.

[33] Z. Liu, K. Shi, N. Chen, Coreference-aware dialogue summarization, in: Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, ACL, Singapore and Online, 2021, pp. 509–519, URL https://aclanthology.org/2021.sigdial-1.53.

[34] M. Peyrard, A simple theoretical model of importance for summarization, in: ACL, ACL, Florence, Italy, 2019, pp. 1059–1073, http://dx.doi.org/10.18653/v1/P19-1101, URL https://aclanthology.org/P19-1101.

[35] K. Riedhammer, B. Favre, D. Hakkani-Tür, A keyphrase based approach to interactive meeting summarization, in: SLT, IEEE, 2008, pp. 153–156.

[36] J.J. Koay, A. Roustai, X. Dai, D. Burns, et al., How domain terminology affects meeting summarization performance, in: COLING, COLING, Barcelona, Spain (Online), 2020, pp. 5689–5695, http://dx.doi.org/10.18653/v1/2020.coling-main.499, URL https://aclanthology.org/2020.coling-main.499.

[37] L. Zhao, W. Xu, J. Guo, Improving abstractive dialogue summarization with graph structures and topic words, in: COLING, COLING, Barcelona, Spain (Online), 2020, pp. 437–449, http://dx.doi.org/10.18653/v1/2020.coling-main.39, URL https://aclanthology.org/2020.coling-main.39.

[38] C.-S. Wu, L. Liu, W. Liu, P. Stenetorp, et al., Controllable abstractive dialogue summarization with sketch supervision, in: ACL-IJCNLP 2021, ACL, Online, 2021, pp. 5108–5122, http://dx.doi.org/10.18653/v1/2021.findings-acl.454, URL https://aclanthology.org/2021.findings-acl.454.

[39] G. Frisoni, G. Moro, L. Balzani, Text-to-text extraction and verbalization of biomedical event graphs, in: COLING, COLING, Gyeongju, Republic of Korea, 2022, pp. 2692–2710, URL https://aclanthology.org/2022.coling-1.238.

[40] G. Domeniconi, G. Moro, R. Pasolini, C. Sartori, Iterative refining of category profiles for nearest centroid cross-domain text classification, in: A.L.N. Fred, J.L.G. Dietz, D. Aveiro, K. Liu, J. Filipe (Eds.), Knowledge Discovery, Knowledge Engineering and Knowledge Management - 6th International Joint Conference, IC3K 2014, Rome, Italy, October 21-24, 2014, Revised Selected Papers, in: Communications in Computer and Information Science, 553, Springer, 2014, pp. 50–67, http://dx.doi.org/10.1007/978-3-319-25840-9_4.

[41] G. Domeniconi, G. Moro, R. Pasolini, C. Sartori, Cross-domain text classification through iterative refining of target categories representations, in: A.L.N. Fred, J. Filipe (Eds.), KDIR 2014 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Rome, Italy, 21 - 24 October, 2014, SciTePress, 2014, pp. 31–42, http://dx.doi.org/10.5220/0005069400310042.

[42] Y. Zhang, Z. Gan, K. Fan, Z. Chen, et al., Adversarial feature matching for text generation, in: D. Precup, Y.W. Teh (Eds.), ICML, in: Proceedings of Machine Learning Research, vol. 70, PMLR, 2017, pp. 4006–4015, URL http://proceedings.mlr.press/v70/zhang17b.html.

[43] E. Jang, S. Gu, B. Poole, Categorical reparameterization with gumbel-softmax, in: ICLR, OpenReview.net, 2017, URL https://openreview.net/forum?id=rkE3y85ee.

[44] M. Firdaus, A.P. Shandeelya, A. Ekbal, More to diverse: Generating diversified responses in a task oriented multimodal dialog system, PLoS One 15 (2020) URL https://api.semanticscholar.org/CorpusID:226269766.

[45] J. Gu, D.J. Im, V.O. Li, Neural machine translation with gumbel-greedy decoding, in: S.A. McIlraith, K.Q. Weinberger (Eds.), Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, AAAI Press, 2018, pp. 5125–5132, http://dx.doi.org/10.1609/AAAI.V32I1.12016.

[46] W. Kool, H. van Hoof, M. Welling, Stochastic beams and where to find them: The gumbel-top-k trick for sampling sequences without replacement, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, in: Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 3499–3508, URL http://proceedings.mlr.press/v97/kool19a.html.

[47] C. Su, H. Huang, S. Shi, P. Jian, X. Shi, Neural machine translation with gumbel tree-lstm based encoder, J. Vis. Commun. Image Represent. 71 (2020) 102811, http://dx.doi.org/10.1016/J.JVCIR.2020.102811.

[48] S. Havrylov, I. Titov, Emergence of language with multi-agent games: Learning to communicate with sequences of symbols, in: I. Guyon, U. von Luxburg, S. Bengio, H.M. Wallach, R. Fergus, S.V.N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 2149–2159, URL https://proceedings.neurips.cc/paper/2017/hash/70222949cc0db89ab32c9969754d4758-Abstract.html.

[49] Y. Chen, Y. Liu, L. Chen, Y. Zhang, DialogSum: A real-life scenario dialogue summarization dataset, 2021, arXiv preprint arXiv:2105.06762.

[50] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, S. Niu, Dailydialog: A manually labelled multi-turn dialogue dataset, 2017, arXiv preprint arXiv:1710.03957.

[51] K. Sun, D. Yu, J. Chen, D. Yu, Y. Choi, C. Cardie, Dream: A challenge data set and models for dialogue-based reading comprehension, Trans. Assoc. Comput. Linguist. 7 (2019) 217–231.

[52] L. Cui, Y. Wu, S. Liu, Y. Zhang, M. Zhou, Mutual: A dataset for multi-turn dialogue reasoning, 2020, arXiv preprint arXiv:2004.04494.

[53] S. Bird, E. Klein, E. Loper, Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, O'Reilly Media, Inc., 2009.

[54] Y. Liu, M. Ott, N. Goyal, J. Du, et al., RoBERTa: A robustly optimized BERT pretraining approach, 2019, CoRR abs/1907.11692. URL http://arxiv.org/abs/1907.11692. arXiv:1907.11692.

[55] P. He, X. Liu, J. Gao, W. Chen, Deberta: decoding-enhanced bert with disentangled attention, in: ICLR, OpenReview.net, 2021, URL https://openreview.net/forum?id=XPZIaotutsD.

[56] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners, 2019.

[57] S. Zhang, S. Roller, N. Goyal, M. Artetxe, et al., OPT: open pre-trained transformer language models, 2022, http://dx.doi.org/10.48550/arXiv.2205.01068, CoRR abs/2205.01068. arXiv:2205.01068.

[58] H.W. Chung, L. Hou, S. Longpre, B. Zoph, et al., Scaling instruction-finetuned language models, 2022, http://dx.doi.org/10.48550/arXiv.2210.11416, CoRR abs/2210.11416. arXiv:2210.11416.

[59] J. Chen, D. Yang, Structure-aware abstractive conversation summarization via discourse and action graphs, in: NAACL, ACL, Online, 2021, pp. 1380–1391, http://dx.doi.org/10.18653/v1/2021.naacl-main.109, URL https://aclanthology.org/2021.naacl-main.109.

[60] K.-h. Huang, S. Singh, X. Ma, W. Xiao, et al., SWING: Balancing coverage and faithfulness for dialogue summarization, in: EACL, ACL, Dubrovnik, Croatia, 2023, pp. 512–525, URL https://aclanthology.org/2023.findings-eacl.37.

[61] Q. Jia, Y. Liu, H. Tang, K. Zhu, Post-training dialogue summarization using pseudo-paraphrasing, in: Findings of the Association for Computational Linguistics: NAACL 2022, Association for Computational Linguistics, Seattle, United States, 2022, pp. 1660–1669, http://dx.doi.org/10.18653/v1/2022.findings-naacl.125, URL https://aclanthology.org/2022.findings-naacl.125.

[62] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, ACL, Barcelona, Spain, 2004, pp. 74–81.

[63] G. Moro, L. Ragazzi, L. Valgimigli, Carburacy: Summarization models tuning and comparison in eco-sustainable regimes with a novel carbon-aware accuracy, in: AAAI, AAAI Press, 2023, pp. 1–9.

[64] T. Zhang, V. Kishore, F. Wu, K.Q. Weinberger, et al., BERTScore: Evaluating text generation with BERT, in: ICLR, OpenReview.net, 2020, URL https://openreview.net/forum?id=SkeHuCVFDr.

[65] W. Yuan, G. Neubig, P. Liu, BARTScore: Evaluating generated text as text generation, in: M. Ranzato, A. Beygelzimer, Y.N. Dauphin, P. Liang, J.W. Vaughan (Eds.), NeurIPS, 2021, pp. 27263–27277, URL https://proceedings.neurips.cc/paper/2021/hash/e4d2b6e6fdeca3e60e0f1a62fee3d9dd-Abstract.html.

[66] S. Narayan, S.B. Cohen, M. Lapata, Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization, in: EMNLP, ACL, Brussels, Belgium, 2018, pp. 1797–1807, http://dx.doi.org/10.18653/v1/D18-1206, URL https://aclanthology.org/D18-1206.

[67] A. Fabbri, I. Li, T. She, S. Li, et al., Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model, in: ACL, ACL, Florence, Italy, 2019, pp. 1074–1084, http://dx.doi.org/10.18653/v1/P19-1102, URL https://aclanthology.org/P19-1102.

[68] W.X. Zhao, K. Zhou, J. Li, T. Tang, et al., A survey of large language models, 2023, http://dx.doi.org/10.48550/arXiv.2303.18223, CoRR abs/2303.18223. arXiv:2303.18223.

[69] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E.H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, W. Fedus, Emergent abilities of large language models, Trans. Mach. Learn. Res. 2022 (2022) URL https://openreview.net/forum?id=yzkSU5zdwD.

[70] R. Schaeffer, B. Miranda, S. Koyejo, Are emergent abilities of large language models a mirage?, CoRR abs/2304.15004 (2023) URL https://doi.org/10.48550/arXiv.2304.15004.

[71] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, et al., Sparks of artificial general intelligence: Early experiments with GPT-4, 2023, http://dx.doi.org/10.48550/arXiv.2303.12712, CoRR abs/2303.12712. arXiv:2303.12712.

[72] F. Gilardi, M. Alizadeh, M. Kubli, ChatGPT outperforms crowd-workers for text-annotation tasks, 2023, http://dx.doi.org/10.48550/arXiv.2303.15056, CoRR abs/2303.15056. arXiv:2303.15056.

[73] J. Wang, Y. Liang, F. Meng, H. Shi, et al., Is ChatGPT a good NLG evaluator? A preliminary study, 2023, http://dx.doi.org/10.48550/arXiv.2303.04048, CoRR abs/2303.04048. arXiv:2303.04048.

[74] M.G. Kendall, A new measure of rank correlation, Biometrika 30 (1/2) (1938) 81–93.

[75] G. Moro, L. Valgimigli, Efficient self-supervised metric information retrieval: A bibliography based method applied to COVID literature, Sensors 21 (19) (2021) 6430, http://dx.doi.org/10.3390/S21196430.

[76] G. Domeniconi, M. Masseroli, G. Moro, P. Pinoli, Discovering new gene functionalities from random perturbations of known gene ontological annotations, in: A.L.N. Fred, J. Filipe (Eds.), KDIR 2014 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Rome, Italy, 21 - 24 October, 2014, SciTePress, 2014, pp. 107–116, http://dx.doi.org/10.5220/0005087801070116.

[77] G. Domeniconi, M. Masseroli, G. Moro, P. Pinoli, Cross-organism learning method to discover new gene functionalities, Comput. Methods Programs Biomed. 126 (2016) 20–34, http://dx.doi.org/10.1016/J.CMPB.2015.12.002.

[78] M. Zhong, Y. Liu, Y. Xu, C. Zhu, et al., DialogLM: Pre-trained model for long dialogue understanding and summarization, in: AAAI, AAAI Press, 2022, pp. 11765–11773, URL https://ojs.aaai.org/index.php/AAAI/article/view/21432.

[79] A. Paszke, S. Gross, F. Massa, A. Lerer, et al., PyTorch: An imperative style, high-performance deep learning library, in: H.M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E.B. Fox, R. Garnett (Eds.), NeurIPS, 2019, pp. 8024–8035, URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

[80] T. Wolf, L. Debut, V. Sanh, J. Chaumond, et al., Transformers: State-of-the-art natural language processing, in: EMNLP, ACL, Online, 2020, pp. 38–45, http://dx.doi.org/10.18653/v1/2020.emnlp-demos.6, URL https://aclanthology.org/2020.emnlp-demos.6.

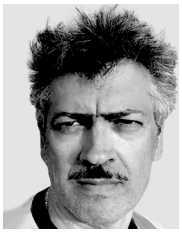
[81] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: ICLR, OpenReview.net, 2019, URL https://openreview.net/forum?id=Bkg6RiCqY7.

[82] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, et al., LoRA: Low-rank adaptation of large language models, in: ICLR, OpenReview.net, 2022, URL https://openreview.net/forum?id=nZeVKeeFYf9.

[83] G. Frisoni, A. Carbonaro, G. Moro, A. Zammarchi, M. Avagnano, NLG-metricverse: An end-to-end library for evaluating natural language generation, in: COLING, COLING, Gyeongju, Republic of Korea, 2022, pp. 3465–3479, URL https://aclanthology.org/2022.coling-1.306.

Paolo Italiani received the B.S. and M.S. degrees in statistical sciences from the University of Bologna, Italy, in 2019 and 2022, respectively. In 2021 he worked 6 months as a data scientist for Qarik Group, where he helped the company enhance its Document Ingestion expertise. His current research interests include semantic parsing, knowledge representation learning, natural language processing and understanding. He received the International Conference on Data Science, Technology and Applications Best Student Paper Award in 2022. in November 2022, Dr. Italiani started pursuing a Ph.D. degree at the Department of Computer Science and Engineering, University of Bologna.

Giacomo Frisoni received the B.S. and M.S. degrees in computer science and engineering from the University of Bologna, Italy, in 2017 and 2020–both with honors. He is currently a third-year Ph.D. student at the Department of Computer Science and Engineering, University of Bologna. His research interests include natural language understanding, large language models, and graph neural networks. He presented several original papers to international journals and peer-reviewed conferences—including top-tier venues like EMNLP, AAAI, and COLING, winning two Best Paper Awards. In 2020, he was among the worldwide selected program attendees at the Cornell, Maryland, Max Planck Pre-doctoral Research School. In the same year, he received the con.Scienze Award for writing one of the ten best Italian scientific research works during the master's thesis. In September–December 2022, he was a visiting postgraduate researcher at the University of Glasgow, School of Computing Science, Scotland. Since 2022, he has been an HuggingFace and Streamlit Student Ambassador.



Gianluca Moro received the Ph.D. degree in computer science and engineering from the Department of Electronics, Computer Science and Systems of the University of Bologna, Italy, in 1999. He is associate professor of text mining, data mining and big data analytics at the Department of Computer Science and Engineering of the University of Bologna and head of the research unit in text mining and natural language processing of the Cesena campus. He co-organized several editions of workshops at VLDB and AAMAS, edited five international books and published more than ninety papers, even in top international conferences such as AAAI, IJCAI, EMNLP, ACL, AAMAS, COLING, etc., also winning several best paper awards. He has led national and international projects on NLP, data mining and machine learning research topics and collaborates with several public and private research organizations.

Antonella Carbonaro received the Ph.D. degree in Intelligent Artificial Systems from the Faculty of Engineering of the University of Ancona, Italy. She won post-doc research grants on Artificial Intelligence. Since 2000 she has been first a researcher, then an associate professor at the Department of Computer Science - Science and Engineering - of the University of Bologna, Italy. Her research is on data and knowledge modeling for the representation of entity semantics and relations, also in the domain of health data. She co-authored scientific publications in international conferences, workshops, books, journals, co-edited books and has served in numerous program committees among international conferences, workshops and as journal referee. She is the leader of the WP Technology and Analytics of the project - "DARE - digital lifelong prevention", supported from Italian Ministry of University and Research (CUP: B53C22006240001).

Claudio Sartori has been full professor of "Systems for Information Processing" in the Department of Computer Science and Engineering of the University of Bologna since 2001. He participated in several Italian and international research projects and carried out also applied research projects in cooperation with industries. For more than ten years he coordinated university programs in the areas of Statistic Sciences and Computer Engineering. Doing research from 1984, he is the author of more than 100 refereed scientific publications and cooperates with international institutions, such as the University Federico Santamaria, Valparaiso, Chile, and the University of Nice - Sophia Antipolis. He was the advisor of seven Ph.D. students and more than a hundred graduated students for the master thesis. His current research interests include Machine Learning, Data Mining, Quantum Machine Learning. He directs a master program in Data Science and Business Analytics for the Bologna Business School.