







# Propose, Assess, Search: Harnessing LLMs for Goal-Oriented Planning in Instructional Videos

Md Mohaiminul Islam<sup>1,2\*</sup>  Tushar Nagarajan<sup>1</sup>  Huiyu Wang<sup>1</sup>  
Fu-Jen Chu<sup>1</sup>  Kris Kitani<sup>1</sup>  Gedas Bertasius<sup>2</sup>  Xitong Yang<sup>1</sup> 

<sup>1</sup>FAIR, Meta      <sup>2</sup>UNC Chapel Hill

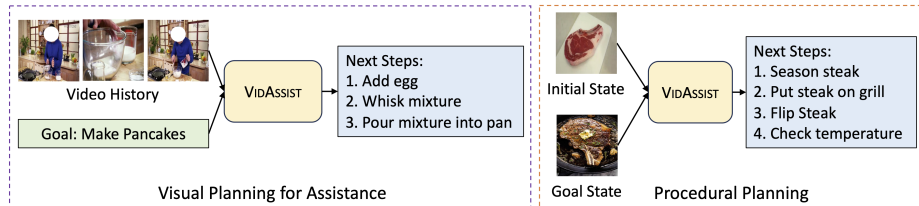
**Abstract.** Goal-oriented planning, or anticipating a series of actions that transition an agent from its current state to a predefined objective, is crucial for developing intelligent assistants aiding users in daily procedural tasks. The problem presents significant challenges due to the need for comprehensive knowledge of temporal and hierarchical task structures, as well as strong capabilities in reasoning and planning. To achieve this, prior work typically relies on extensive training on the target dataset, which often results in significant dataset bias and a lack of generalization to unseen tasks. In this work, we introduce VIDASSIST, an integrated framework designed for zero/few-shot goal-oriented planning in instructional videos. VIDASSIST leverages large language models (LLMs) as both the knowledge base and the assessment tool for generating and evaluating action plans, thus overcoming the challenges of acquiring procedural knowledge from small-scale, low-diversity datasets. Moreover, VIDASSIST employs a breadth-first search algorithm for optimal plan generation, in which a composite of value functions designed for goal-oriented planning is utilized to assess the predicted actions at each step. Extensive experiments demonstrate that VIDASSIST offers a unified framework for different goal-oriented planning setups, e.g., visual planning for assistance (VPA) and procedural planning (PP), and achieves remarkable performance in zero-shot and few-shot setups. Specifically, our few-shot model outperforms the prior fully supervised state-of-the-art method by +7.7% in VPA and +4.81% PP task on the COIN dataset while predicting 4 future actions.

## 1 Introduction

This work addresses goal-oriented planning, a task that requires forecasting action plans to achieve a given goal based on the current observations. Depending on the specifications of the current observation and the desired goal, the task can be instantiated in two different setups, both of which have been *independently* explored in prior literature: (a) Visual Planning for Assistance (VPA) [16], where the current observation involves an untrimmed video history of the user’s

---

\* Work done during an internship at Meta.



**Fig. 1:** Goal-oriented planning aims to generate action plans to achieve a given goal based on the visual observations. The task unifies two setups that have been *independently* explored in prior literature: Visual Planning for Assistance [16] (left) and Procedural Planning [5, 6, 14, 22, 26, 33] (right).

progress and the goal is described in natural language; and (b) Procedural Planning (PP) [6], where the initial and goal states are presented with images (Figure 1). Both setups are of great interest in developing intelligent embodied systems capable of understanding human decision-making processes and assisting humans in solving procedural tasks, such as cooking or assembling furniture.

Goal-oriented planning can be viewed as a sequential decision-making process [4], in which a goal-conditioned policy sequentially predicts the next action based on the observed states. However, learning such a policy from video data poses significant challenges, as it requires a comprehensive understanding of procedural tasks, including the temporal and hierarchical interconnections between action steps. To achieve this, prior work typically relies on extensive training on target datasets, utilizing full supervision from action labels, visual representations [5, 6, 22], or text description [14, 26, 33]. These methods share a common limitation – procedural knowledge, when derived from strong supervision, exhibits a significant bias towards the training dataset, particularly when the scale of annotation and the diversity of tasks are limited. This hinders the model’s ability to generalize across new data distribution (*e.g.*, action plans of varying granularity) and to adapt to unseen tasks. Note that the capability for zero/few-shot learning in a planning system is essential for its deployment within intelligent embodied assistants or robots [8, 23].

In this paper, we propose VIDASSIST, an integrated framework that tackles goal-oriented planning in a zero- or few-shot learning scenario. Drawing inspiration from human decision-making process [13, 20] and research in robot planning [3, 10], we design a search-based framework that facilitates deliberate decision-making at each prediction step and ultimately generates the optimal action plan based on search heuristics. Furthermore, our framework leverages pretrained large language models (LLMs) for generating and evaluating action plans, serving as the knowledge base and the tool for assessment. We hypothesize that LLMs, pretrained with a wealth of Internet-sourced text including web knowledge bases (*e.g.*, wikiHow [1]), inherently possess extensive knowledge about a variety of procedural tasks and robust reasoning capabilities.

The VIDASSIST model first processes visual observations and goals through a Socratic approach [31, 34], *i.e.*, by converting visual inputs into textual descriptions using video/image understanding models. Afterward, we predict the

optimal action plan consisting of the future action steps by three key processes. (i) *Propose* – prompting LLMs to predict  $K$  probable subsequent actions in free-form language, conditioned on the information grounded from visual observations/goals as well as the action plan predicted in prior steps. This allows us to leverage the procedural knowledge embedded within LLMs while also accounting for the inherent uncertainty of procedural tasks. (ii) *Assess* – evaluating each proposed action through a mixture of value functions specifically designed for goal-oriented planning. The value functions assess the coherence and viability of the trajectory of predicted actions utilizing an LLM. They also evaluate the confidence of language generation and its alignment with admissible actions. (iii) *Search* – identifying the optimal action plan based on the assessment scores employing a breadth-first search (BFS) algorithm. Low-scoring action proposals are pruned dynamically to improve the efficiency of the search algorithm from a potentially very large search tree.

The proposed VIDASSIST framework offers enhanced planning capabilities than the standard LLM-prompt-based baseline, particularly for long-horizon planning. On the VPA task, our zero-shot model outperforms the LLM baseline by 12.9% and 6.6% success rate on COIN and CrossTask datasets while predicting 3 future actions (i.e., planning horizon 3). Similarly, on the PP task, zero-shot VIDASSIST achieves 7.41% and 6.62% higher success rates on COIN and CrossTask datasets for a planning horizon 4. Finally, our few-shot model achieves state-of-the-art performance on the COIN dataset, surpassing the prior best method by 7.7% in VPA and 4.81% in PP for a planning horizon 4.

## 2 Related Works

**Planning in instructional videos.** The task of procedural planning (PP) in instructional videos was first introduced in [6], where given the start and the end goal image of a task, the model needs to output intermediate steps to achieve the goal. Since then, many follow-up works have been proposed for the procedural planning task by leveraging supervision from action labels and visual features of the intermediate steps [5, 22]. However, such supervision requires expensive annotation efforts to localize the intermediate steps within a video, which is challenging to extend to large-scale datasets. Recent work instead relies on language representation for supervision [14, 33]. Moreover, recent work [16] has introduced another setting for goal-oriented planning in instruction videos, called visual planning for assistance (VPA), where given the goal and video history of the completed actions, the model needs to output the future steps to complete the task. They proposed a fully supervised language model-based approach for the VPA task. Most prior works in this area developed specialized, fully supervised architectures for different goal-oriented planning tasks (e.g., PP and VPA tasks) in instructional videos. On the contrary, we propose a unified model for various goal-oriented planning tasks (e.g., PP and VPA tasks), which shows strong performance with limited annotated data.

**LLMs for task planning.** Socratic Models [31] pioneered the use of LLMs for multimodal reasoning in a zero-shot/few-shot manner. Inspired by this work, several subsequent works [7, 34] have been proposed for future action anticipation and forecasting from videos. These methods follow a common high-level approach - translating videos into text descriptions using various video understanding models (action recognition, captioning, etc.) and utilizing LLMs to forecast future actions from the translated text descriptions. Furthermore, utilizing language as an intermediate representation allows efficient processing for long-range videos [11, 12, 32]. However, these methods do not incorporate any deliberate planning mechanism specifically tailored for procedural videos. On the other hand, several works [10, 21, 29] in the robotics domain have explored generating action plans for robots and general problem-solving utilizing deliberate planning mechanisms. Drawing motivation from both streams of research, we have developed a unified framework for goal-oriented planning in instructional videos. Our model leverages a Socratic approach for visual understanding and incorporates a deliberate planning mechanism using the introduced propose, assess, and search techniques.

### 3 Methodology

In this section, we provide a detailed description of our framework for goal-oriented planning in instructional videos. We first present a baseline method in Section 3.2, which leverages LLMs to generate action plans. The baseline demonstrates promising results in anticipating short-horizon actions, but its performance significantly diminishes for planning over a long horizon. In Section 3.3, we introduce our framework, VIDASSIST, which combines LLMs with a search-based algorithm to achieve superior results in long-horizon planning.

#### 3.1 Task formulation

Given the current observation  $\mathcal{O}$  and the goal  $\mathcal{G}$ , *goal-oriented planning* requires the model to generate an action plan, *i.e.*, a sequence of actions  $\mathcal{A} = \{a_1, \dots, a_T\}$ , that transforms the current state to the goal state, where  $T$  is the planning horizon. Here, each action  $a_t \in \mathbb{R}^C$  is a categorical label with corresponding text description (*e.g.*, "break eggs"), and  $C$  indicates the total number of admissible actions. Moreover, we focus on zero-shot and few-shot setups since they are more practical in embodied AI applications [8, 23]. Only action supervision  $\mathcal{A}$  is available during training, and it is limited to  $N$  samples per task ( $N = 0$  and 10 for zero-shot and few-shot learning, respectively).

In real-world scenarios, both  $\mathcal{O}$  and  $\mathcal{G}$  can take various forms, including video, image, or text. For example, a user can describe their desired goal in language (*e.g.*, "I want to make pancakes"), or simply show an image describing the goal state (*e.g.*, and image of cooked steak, as shown in Figure 1). In this paper, we consider the following two types of goal-oriented planning.

**Visual Planning for Assistance (VPA)** [16]. In this setup, the observation  $\mathcal{O}$  is presented as an untrimmed video history capturing the user’s progress:  $\mathcal{O} = \{v_{-H}, \dots, v_0\}$ , where  $H$  denotes the number of clips. The user’s goal is specified through natural language description  $\mathcal{G} = \{d\}$ .

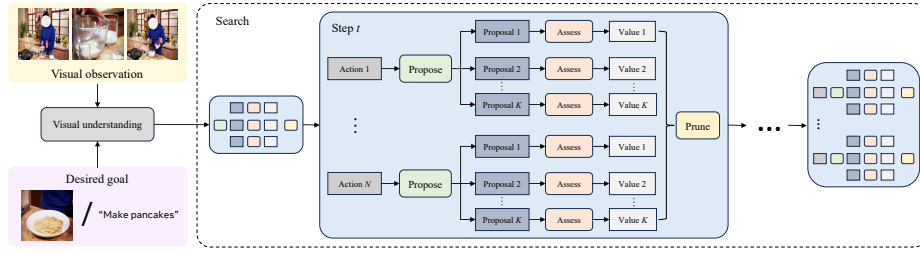
**Procedural Planning (PP)** [5, 6, 14, 22, 26, 33]. In this setup, both the initial and goal state is specified with an image:  $\mathcal{O} = \{v_0\}$ ,  $\mathcal{G} = \{v_T\}$ .

### 3.2 Baseline: LLM-based goal-oriented planning

Prior research has shown that large language models (LLMs), pre-trained on colossal amounts of Internet-sourced data, inherently possess extensive world knowledge which enables them to engage in problem solving and reasoning without the need for additional training [3, 10, 30, 31]. To investigate how LLMs can be applied for goal-oriented planning in instructional videos, we develop a baseline system that first converts visual observations  $\mathcal{O}$  and goals  $\mathcal{G}$  into text descriptions and then prompts LLMs to generate action plans accordingly.

**Understanding visual observations and goals.** We consider the Socratic model [31], a predominant approach for visual understanding and grounding, to represent visual observations and goals for LLM-based planning. The main idea is to convert non-textual modalities (*e.g.*, images, videos) into text for downstream LLMs. Specifically, for visual observations presented as video history, we follow the prior work [16] to divide the raw video into fixed-length window clips [11] and predict the action category of each clip using an action recognition model  $F_{\text{video}}$  [28]. If consecutive clips have the same predicted action category, we consolidate them into one prediction. Consequently, we create an action history sequence  $\tilde{\mathcal{O}} = F_{\text{video}}(\{v_{-H}, \dots, v_0\}) = \{\tilde{a}_{-H}, \dots, \tilde{a}_0\}$  comprising textual descriptions of predicted actions. For visual observations/goals that are specified with images, we use an image-based step classifier  $F_{\text{image}}$  [14] to predict the initial state  $\tilde{\mathcal{O}} = F_{\text{image}}(\{v_0\}) = \{\tilde{a}_0\}$  and the goal state  $\tilde{\mathcal{G}} = F_{\text{image}}(\{v_T\}) = \{\tilde{a}_T\}$ , represented in textual description. Note that in VPA, the desired goal is already provided through a language description, thus eliminating the need for visual understanding:  $\tilde{\mathcal{G}} = \mathcal{G} = \{d\}$ .

**LLM-based next-action prediction.** Inspired by prior work on LLM-based task planning for embodied AI [3, 10], we prompt a state-of-the-art LLM (*e.g.*, Llama-2 [25]) to predict the most probable subsequent action, taking advantage of the procedural knowledge embedded within the pretrained model. We employ Llama-2 since it is the best open-source LLM for diverse tasks; however, our framework can easily incorporate other LLMs. Rather than generating the entire action plan in one run, we employ an autoregressive approach to generate subsequent actions step-by-step until the given goal is achieved. As shown in Figure 3, our LLM prompt includes (i) the initial observation  $\tilde{\mathcal{O}}$ , (ii) the goal state  $\tilde{\mathcal{G}}$ , and (iii) the predicted action plan up to the current step  $t$ :  $\mathcal{A}_t = \{\hat{a}_1, \dots, \hat{a}_t\}$ , all in text format. In the few-shot setup, we augment the prompt by appending a set of in-context examples containing goals and their corresponding action plans.



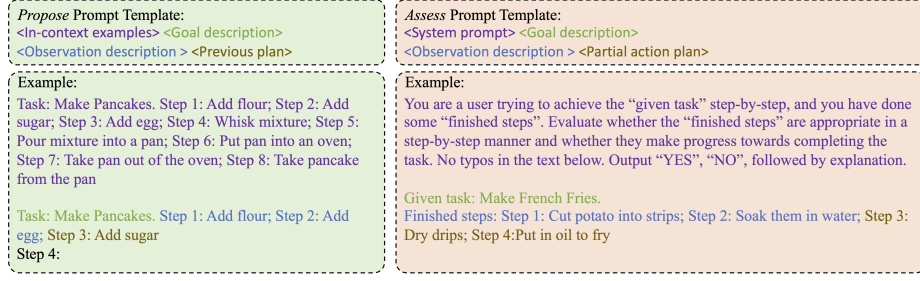
**Fig. 2:** Overview of the VIDASSIST framework. We first process visual observations and goals by transforming the visual inputs into textual descriptions using visual understanding models. Then, we leverage a search-based approach for optimal plan generation: at each step, we *propose*  $K$  probable subsequent actions and *assess* them using a composite of value functions specifically designed for goal-oriented planning. LLMs are employed as both the knowledge base and the assessment tool in the search process, and we illustrate the details in Fig. 3.

One challenge arises when naively employing LLM-based predictions as the action plan: the output space of LLM is unconstrained, and the prediction expressed in free-form language often cannot be translated to admissible actions. This leads to suboptimal action plans, especially for predicting longer horizon action steps. To address this issue, similar to the previous studies [10, 14], we map the free-form LLM output to the most semantically similar admissible action using an off-the-shelf text embedding model (*e.g.*, Sentence-BERT [18]). The admissible action that exhibits the highest cosine similarity with the LLM output will then be included in the predicted action plan.

### 3.3 VIDASSIST

While the baseline method already demonstrates promising planning results in zero-/few-shot settings, especially for short-term anticipation, its performance diminishes significantly for planning over a long horizon. We hypothesize that deliberate planning mechanisms to explore a combinatorial space of partial solutions and assessment of partial plans with problem-specific heuristic functions are essential to finding the optimal action plan for the procedural tasks [15, 19].

We propose VIDASSIST, a search-based framework for optimal plan generation. Figure 2 presents an overview of our framework. At each step, we first *propose*  $K$  probable subsequent actions using the method described for the baseline in Section 3.2. We sample  $K$  times from the LLM with the same prompt to capture the uncertainty inherent to the procedural tasks [2, 33]. Then, we *assess* each proposed action using a composite of value functions specifically tailored for the task of goal-oriented planning. The value functions entail evaluating the coherence and viability of the entire trajectory of predicted actions, leveraging procedural knowledge inherent in the LLM or a task graph derived from few-shot examples. It is noteworthy that this assessment process is crucial to our framework, as it facilitates informed and deliberate decision making [29, 30] and extends beyond the conventional token-level generation process typically used



**Fig. 3:** Templates and examples of the LLM prompts we design for subsequent action proposal (left) and partial plan evaluation (right).

in LLM inference. Finally, we employ a breadth-first *search* (BFS) algorithm to identify the optimal action plan according to the assessment scores. A set of the  $\tilde{K}$  most promising actions per step is maintained for future evaluation ( $\tilde{K} < K$ ), while the others are pruned to enhance efficiency. Once the best-predicted action at step,  $T$  is determined, the optimal action plan can be obtained by backtracking the sequence of predicted actions that led to the generation of  $\hat{a}_T^*$ . Below, we detail the value functions that contribute to determining the assessment score.

Let us denote the  $K$  sampled outputs of the LLM during the  $t$ -th “propose” step as  $\{s_{t,1}, \dots, s_{t,K}\}$ , and the projected admissible actions are  $\{\hat{a}_{t,1}, \dots, \hat{a}_{t,K}\}$  (Section 3.2). Our assessment score is a weighted sum of four components, each evaluating a different aspect of the action plan:

**Text generation score.** This function evaluates the probability of an LLM generating a given description. Intuitively, descriptions of unlikely subsequent actions will yield lower scores. Similar to [10], we compute the mean log probability of the proposed action descriptions as their likelihood scores:

$$V_G^k = \frac{1}{n_k} \sum_{i=1}^{n_k} \log p_\theta(s_{k,i} | \tilde{s}_{k,<i}), \quad (1)$$

where  $n_k$  is the number of tokens in  $s_k$ ,  $\theta$  parameterizes the LLM, and  $\tilde{s}_{k,<i}$  denotes the concatenation of the prompt with previously generated tokens.

**Text mapping score.** This function assesses the confidence of mapping a free-form description generated by the LLM to an admissible action. Since this mapping is achieved by identifying the admissible action with the highest cosine similarity to the generated description, we use this similarity value as the confidence score:

$$V_M^k = \max_{\hat{a}_k} \frac{\text{Emb}(s_k) \cdot \text{Emb}(\hat{a}_k)}{\|\text{Emb}(s_k)\| \|\text{Emb}(\hat{a}_k)\|}, \quad (2)$$

where  $\text{Emb}(\cdot)$  is a text embedding function (e.g., Sentence-BERT [18]), and  $\hat{a}_k$  the admissible action corresponding to highest similarity.

**Partial plan evaluation.** This function evaluates the coherence and viability of a partially generated plan up to step  $t$ . In particular, we repurpose LLMs as the assessment tool by creating a dedicated prompt that requests the LLMs to assess



the soundness of the partial plan in order to achieve the given goal (Figure 3). Unlike the text generation score that emphasizes token-level evaluation, this function evaluates the *semantic* action phrases and takes into account the entire prediction trajectory. This allows the LLM to self-evaluate the progress it makes towards solving the planning problem through a deliberate reasoning process [29, 30]. To get the evaluation score, we extract the logits of the “YES” and “NO” tokens from the output of the LLM, followed by softmax normalization:

$$V_P^k = e^{L_\theta(\langle \text{YES} \rangle | \tilde{p})} / (e^{L_\theta(\langle \text{YES} \rangle | \tilde{p})} + e^{L_\theta(\langle \text{NO} \rangle | \tilde{p})}), \quad (3)$$

Here,  $L_\theta$  denotes the logit values generated by the LLM, and  $\tilde{p}$  denotes the prompt, including the partially generated plan (Figure 3).

**Few-shot task graph.** This function provides task-specific prior knowledge to our assessment method. In particular, we construct a first-order task graph from the few-shot examples by counting the frequencies of the consecutive action step pairs. Then, we assign a probability value to a partially generated plan  $\mathcal{A}_t = \{\hat{a}_1, \dots, \hat{a}_t\}$  using the following equation. Note that we do not use this value function for zero-shot setup.

$$V_{TG}^k = \prod_{i=1}^{i=t} G_\theta(\hat{a}_{t,i} | \hat{a}_{t,i-1}), \quad (4)$$

Here,  $G_\theta$  is the few-shot task graph which captures the transitional probabilities of different actions, and  $G_\theta(\hat{a}_{t,i} | \hat{a}_{t,i-1}) = f(\hat{a}_{i-1}, \hat{a}_i) / \sum_{a_y} f(\hat{a}_{i-1}, \hat{a}_y)$ , where  $f(\hat{a}_x, \hat{a}_y)$  is the frequencies of the consecutive actions steps  $(\hat{a}_x, \hat{a}_y)$  calculated from the few-shot examples.

### 3.4 Implementation Details

We use the Llama-2-70B [25] as our default Large Language Model. In the visual planning for assistance task, we segment the video history into 1-second clips and utilize VideoCLIP [28] to predict the action category of each clip, following prior works [16]. Conversely, for the procedural planning task, we adopt the CLIP [17]-based double retrieval model to predict the initial and goal states from the given images, following the previous work [14]. In the few-shot setup, we incorporate 3 in-context examples from the training set for the VPA task and 10 in-context examples from the training set for the PP task.

## 4 Experimental Setup

### 4.1 Datasets

We conduct our experiments on two widely used instructional video datasets, COIN [24] and CrossTask [35]. COIN is a large-scale dataset containing 11,827 videos with 180 different procedural tasks. On the other hand, CrossTask contains 2,750 videos from 18 procedural tasks. We use the standard train/val/test splits and follow the preprocessing steps of prior works to generate samples with input-output pairs for both VPA [16] and PP [6] tasks.



**Table 1: Visual Planning for Assistance on COIN.** VIDASSIST achieves the best performance in zero- and few-shot (3 examples) setups. Moreover, our model outperforms the fully supervised state-of-the-art model VLaMP [16] by 7.6%/3.5%/4.8% SR for predicting T=1/3/4 future steps using only 3 in-context examples.

Method	T=1	T=3			T=4		
	mAcc	SR	mAcc	mIoU	SR	mAcc	mIoU
<b>Fully Supervised</b>							
Most Probable	0.7	1.6	4.3	6.8	1.6	8.2	15.3
Most Probable w/ goal	23.9	10.9	18.0	24.9	9.1	16.3	32.2
DDN [6]	29.3	10.1	22.3	32.2	7.0	21.0	37.3
VLaMP [16]	45.2	18.3	39.2	56.6	9.0	35.2	54.2
<b>Zero-Shot</b>							
Random	0.1	0.0	0.1	0.2	0.0	0.1	0.2
Random w /goal	24.5	1.7	21.4	42.7	0.3	20.1	47.7
LLM Baseline [25]	28.5	2.4	25.8	43.6	0.7	26.1	45.5
LLM Agent Baseline [10]	31.1	3.1	27.1	45.5	1.1	28.5	48.0
VIDASSIST	44.5	15.3	38.0	56.9	6.1	30.6	57.0
<b>Few-Shot</b>							
LLM Baseline [25]	36.8	10.2	36.6	50.8	6.1	30.5	51.5
LLM Agent Baseline [10]	38.2	11.1	40.6	52.8	6.8	33.5	53.5
VIDASSIST	<b>52.8</b>	<b>21.8</b>	<b>44.4</b>	<b>64.4</b>	<b>13.8</b>	<b>38.3</b>	<b>66.3</b>

## 4.2 Evaluation Protocol

**Evaluation Metrics.** Following prior works [6, 16], we use three evaluation metrics. **Success Rate (SR)** considers an action plan successful only if the predicted action steps and the order match exactly with the ground truth. **Mean Accuracy (mAcc)** evaluates the accuracy of individual predicted action steps of a procedural plan compared to their corresponding ground-truth steps at each timestamp. **Mean Intersection Over Union (mIoU)** considers predicted and ground-truth action plans as two sets and measures the overlap between them.

**Planning Horizon.** For the VPA task, evaluation is conducted for planning horizons of T=1, 3, and 4, where the model outputs T future steps given the video history and goal. For the PP task, evaluation is performed for planning horizons of T=3 and 4, where the model outputs the start step, end step, and the T-2 intermediate steps given the task name, start and goal images of a task.

## 4.3 Baselines

**Fully Supervised Baselines.** For the VPA task, we compare with prior state-of-the-art supervised approaches DDN [6] and VLaMP [16]. Following prior works [16], we also compare with a *most probable* baseline, which selects the most probable next action given the previous action, and a *most probable w/ goal* baseline, which chooses the most probable action given the previous action from the actions of a particular task. These two baselines obtain such conditional probabilities by counting frequencies from the training set. On the other hand,

**Table 2: Visual Planning for Assistance on CrossTask.** Our model performs best in zero- and few-shot setups (3 examples). VIDASSIST outperforms the fully supervised state-of-the-art model VLaMP [16] by 3.0% SR while predicting T=4 future steps.

Method	T=1	T=3		T=4			
	mAcc	SR	mAcc	mIOU	SR	mAcc	mIOU
<b>Fully Supervised</b>							
Most Probable	10.4	1.7	6.1	9.9	1.3	5.5	13.9
Most Probable w/ goal	12.4	2.4	8.9	15.5	1.5	7.9	20.5
DDN [6]	33.4	6.8	25.8	35.2	3.6	24.1	37.0
LTA [9]	26.9	2.4	24.0	35.2	1.2	21.7	36.8
VLaMP [16]	<b>50.6</b>	10.3	35.3	44.0	4.4	31.7	43.4
<b>Zero-Shot</b>							
Random	0.9	0.0	0.9	1.5	0.0	0.9	1.9
Random w /goal	13.2	0.3	13.4	23.6	0.0	12.7	27.8
LLM Baseline [25]	25.8	2.1	23.2	27.7	0.4	18.9	33.0
LLM Agent Baseline [10]	28.7	3.1	25.6	29.9	0.8	20.0	35.6
VIDASSIST	38.7	8.7	28.5	44.1	4.6	25.8	46.8
<b>Few-Shot</b>							
LLM Baseline [25]	31.7	4.6	29.7	35.6	1.1	22.2	41.3
LLM Agent Baseline [10]	33.1	5.8	31.3	39.6	2.1	24.7	44.2
VIDASSIST	47.8	<b>12.0</b>	<b>36.7</b>	<b>48.9</b>	<b>7.4</b>	<b>31.9</b>	<b>51.6</b>

for the procedural planning task, we compare against strong supervised methods such as DDN [6], PlaTe [22], Ext-GAIL [5], P3IV [33], E3P [26], and LFP [14].

**Zero-Shot Baselines.** We consider four zero-shot baselines that do not use annotated data for the action plan prediction. (1) A *Random* baseline selects an action randomly from the particular dataset at each step. (2) *Random w/ goal* selects an action randomly from the applicable actions to that goal at each step. (3) *LLM Baseline* [25] described in Section 3.2, provides goal and action history to an LLM as input prompts and outputs the future action plan. (4) *LLM Agent Baseline* [10], an improved version of the LLM Baseline which also generates multiple candidate actions at each step as our model. However, it lacks the deliberate planning mechanism utilizing the propose-assess-based search framework. Both LLM Baseline and LLM Agent Baseline use the same LLM and video history understanding model as VIDASSIST.

**Few-Shot Baselines.** We provide a few in-context examples of action plans from the training set for the *LLM* [25] and the *LLM Agent* [10] baselines. The prompt for the VPA is longer than the PP task because additional action history information is included. Thus, we can accommodate more in-context examples for the PP task (up to 10) while limiting to 3 examples for the VPA task.

## 5 Results and Analysis

### 5.1 Main Results on Visual Planning for Assistance

We conduct experiments on both the COIN [24] and CrossTask [35] datasets for the VPA task, with results presented in Table 1 and Table 2. Our analy-

**Table 3: Procedural Planning on COIN.** VIDASSIST achieves the best performance in zero-shot and few-shot (10 examples) settings. Compared to the fully supervised state-of-the-art model LFP [14], our model achieves significantly better performance (+4.81% SR) for longer horizon T=4 steps.

Method	T=3			T=4		
	SR	mAcc	mIOU	SR	mAcc	mIOU
<b>Fully Supervised</b>						
DDN [6]	13.90	20.19	64.78	11.13	17.71	68.06
P3IV [33]	15.40	21.67	76.31	11.32	18.85	70.53
E3P [26]	19.57	31.42	84.95	13.59	26.72	<b>84.72</b>
PPDP [27]	21.33	45.62	51.82	14.41	44.10	51.39
LFP [14]	<b>30.64</b>	54.72	76.86	15.97	<b>50.70</b>	75.30
<b>Zero-Shot</b>						
Random	0.01	0.01	2.47	0.01	0.01	2.32
LLM Baseline [25]	13.04	45.15	64.45	4.46	38.55	73.96
LLM Agent Baseline [10]	14.15	47.07	67.41	5.46	40.18	75.17
VIDASSIST	18.44	50.63	75.64	9.07	42.72	80.83
<b>Few-Shot</b>						
LLM Baseline	21.79	51.17	68.17	11.25	42.17	76.17
LLM Agent Baseline [10]	23.65	52.55	69.00	12.20	43.19	77.71
VIDASSIST	29.20	<b>54.76</b>	<b>78.02</b>	<b>20.78</b>	49.07	78.93

sis reveals several noteworthy findings. Firstly, our proposed VIDASSIST model demonstrates superior performance across all metrics in both zero-shot and few-shot setups. In the zero-shot scenario, VIDASSIST surpasses the LLM Baseline, by 12.9%/5.4% SR on COIN and 6.6%/4.4% SR on CrossTask for predicting T=3/4 future steps. In the few-shot setting, VIDASSIST outperforms LLM Baseline by 11.6%/7.7% SR on COIN and 7.4%/6.3% SR on CrossTask for T=3/4. This demonstrates the inadequacy of the simple LLM prompt-based techniques for this task, and highlights the effectiveness of the deliberate planning mechanism of our model. Secondly, our model consistently outperforms the LLM Agent baseline by a considerable margin across all metrics in both datasets. Specifically, VIDASSIST achieves a 12.2% higher SR on COIN and 5.6% higher SR on CrossTask in the zero-shot setting for predicting T=3 future steps. This shows the significance of the search-based planning mechanism of our model. Lastly, VIDASSIST outperforms the state-of-the-art fully supervised method VLaMP [16] using only 3 examples by 3.5%/4.8% SR on COIN and 1.7%/3.0% SR on CrossTask for T=3/4 steps. This highlights the superior planning ability of our framework for the VPA task, even with limited annotated samples.

## 5.2 Main Results on Procedural Planning

We conduct experiments on both the COIN [24] and CrossTask [35] datasets for the PP task and show the results in Table 3 and Table 4. Similar to the VPA task, our model performs strongly in both zero-shot and few-shot scenarios. Specifically, in the zero-shot setting, VIDASSIST surpasses the standard LLM

**Table 4: Procedural Planning in CrossTask dataset.** Our model achieves the best performance in both zero- and few-shot setups. It also achieves comparable performance as the state-of-the-art fully supervised models despite using only 10 labeled examples.

Method	T=3			T=4		
	SR	mAcc	mIOU	SR	mAcc	mIOU
<b>Fully Supervised</b>						
DDN [6]	12.18	31.29	47.48	5.97	27.10	48.46
PlaTe [22]	16.00	36.17	65.91	14.00	35.29	55.36
Ext-GAIL [5]	21.27	49.46	73.89	13.40	44.16	70.01
P3IV [33]	23.34	49.96	73.89	13.40	44.16	70.01
E3P [26]	26.40	53.02	74.05	<b>16.49</b>	48.00	70.16
PPDP [27]	26.47	55.35	58.95	15.40	49.42	56.99
LFP	<b>30.55</b>	<b>59.59</b>	<b>76.86</b>	15.97	50.70	<b>75.30</b>
<b>Zero-Shot</b>						
Random	0.01	0.94	1.66	0.01	1.83	1.66
LLM Baseline [25]	9.04	44.30	61.16	3.27	36.61	64.91
LLM Agent Baseline [10]	11.21	47.07	65.98	5.11	38.18	68.66
VIDASSIST	14.60	52.60	68.38	9.89	40.85	70.35
<b>Few-Shot</b>						
LLM Baseline	20.79	53.25	69.96	8.97	44.32	68.81
LLM Agent Baseline	22.67	54.48	70.10	11.06	46.63	70.14
VIDASSIST	28.85	58.12	75.36	15.45	<b>51.51</b>	72.61

baseline, LLM Baseline, by 5.40%/7.41% SR in COIN and 5.56%/6.62% SR in CrossTask datasets for action horizon T=3/4. This highlights the efficacy of our search-based deliberate planning mechanism compared to the standard LLM baseline. Furthermore, our model outperforms the LLM Agent Baseline baseline by 4.29% SR in COIN and 3.39% SR in CrossTask for T=3 steps in zero-shot setups, demonstrating the effectiveness of our explicit planning technique. Finally, compared to the state-of-the-art supervised model LFP, our few-shot (10 examples) VIDASSIST achieves comparable performances, even outperforming in several cases (e.g., +4.81% boost in T=4 steps for the COIN dataset).

### 5.3 Ablation Studies

**Importance of Value Functions.** In this section, we conduct an ablation study to evaluate the significance of four value functions in our goal-oriented planning framework for instructional videos: text generation score, text mapping score, partial plan evaluation, and few-shot task graph (detailed explanations can be found in Section 3.3). The performance of different value functions and their combinations is presented in Table 5. First, we observe that the LLM-based partial plan evaluation function achieves the highest success rate when using only one value function as the assessment method (rows 1-4). Additionally, when using a combination of three value functions, the removal of the partial plan evaluation function results in the largest drop in performance (-4.12%/5.31% SR in VPA/PP for T=3 steps) (rows 5-9). This highlights the efficacy of utilizing

**Table 5: Importance of Value Functions.** Combining all four value functions leads to the best performance in both VPA and PP tasks.

Generation Score	Mapping Score	Task Graph	Partial Plan	VPA			PP	
				T=1	T=3	T=4	T=3	T=4
✓	✓	✓	✓	40.18	11.60	6.70	18.25	11.33
				39.67	10.21	6.10	17.25	11.30
				38.23	9.35	5.50	19.57	13.19
✓	✓	✓	✓	47.10	16.10	9.10	24.44	16.20
				45.66	16.96	9.51	24.38	16.89
				50.67	19.91	12.33	27.81	19.03
				49.69	19.71	12.01	26.76	18.67
✓	✓	✓	✓	<b>52.20</b>	<b>21.08</b>	<b>13.80</b>	<b>29.69</b>	<b>20.78</b>

**Table 6: Performance with Ground-Truth Action History.** Ground-truth action steps yields significantly better performance than predicted actions on COIN, indicating that enhancing the visual models could improve the performance of VIDASSIST.

Plan Horizon (T)	VPA			PP		
	SR	mAcc	mIOU	SR	mAcc	mIOU
1	65.96	65.96	65.96	-	-	-
3	41.33	57.16	75.99	66.13	66.13	66.13
4	25.69	44.16	70.10	38.19	54.17	63.31

LLM as the assessment tool for evaluating action plans in goal-oriented tasks. Moreover, we note that the combination of text generation score, text mapping score, and partial plan evaluation yields the second-highest performance across all metrics. Finally, the combination of all four value functions leads to the highest overall performance, demonstrating the importance of each proposed value function for goal-oriented planning tasks.

**Performance with Ground-Truth Action History.** In this section, we investigate the performance of our model using ground-truth action steps instead of predicted action steps of visual observations. Specifically, we use ground-truth action history and the goal as input prompts for the LLM in the VPA task. On the other hand, for the PP task, we employ ground-truth start and end action steps as input prompts for the LLM. The results in Table 6 indicate that ground-truth observations yield significantly better performance than predicted observations. For the VPA task, VIDASSIST achieves improvements of 13.16%, 19.53%, and 11.89% in SR for T=1, 2, and 3 steps, respectively, when using ground-truth action history instead of predicted action history. On the other hand, in the PP task, the SR improves by 36.93% and 17.41% for T=3 and 4 steps by using ground-truth start and goal steps instead of predicted steps. These findings suggest that improving the visual understanding model can further enhance the performance of our framework.

**Ablation of LLM Sizes.** Table 7 presents the success rate of our few-shot model on the COIN dataset [24] using different LLMs. Firstly, we note that even the smallest Llama-2-7B LLM, VIDASSIST demonstrates good performance. For

**Table 7: Ablation of LLM Sizes.** Utilizing stronger LLMs results in better performance. We show the success rate for different planning horizons (T) for both tasks.

LLM Size	VPA			PP	
	T=1	T=3	T=4	T=3	T=4
Llama-2-7B	48.60	19.61	11.71	26.78	17.72
Llama-2-30B	50.16	20.07	12.28	28.10	18.96
Llama-2-70B	<b>52.20</b>	<b>21.08</b>	<b>13.80</b>	<b>29.69</b>	<b>20.78</b>

**Table 8: Ablation of In-context Examples.** Performance increases with the increase of in-context examples. We could fit upto 3 examples for the VPA task and 10 examples for the PP task in the LLM prompt.

Example Shots	VPA			PP	
	T=1	T=3	T=4	T=3	T=4
0	44.50	15.30	6.10	18.44	9.07
1	48.50	18.17	10.38	21.90	14.13
3	<b>52.20</b>	<b>21.08</b>	<b>13.80</b>	25.25	17.67
10	-	-	-	<b>29.69</b>	<b>20.78</b>

instance, it surpasses the fully supervised state-of-the-art model VLaMP [16] by 1.31% and 2.71% SR for T=3 and 4 in the VPA task while utilizing only 3 examples. Secondly, we observe a consistent improvement in performance with increasing LLM sizes. This indicates that leveraging more powerful LLMs has the potential to further improve the performance of the VIDASSIST framework.

**Ablation of In-context Examples.** Finally, we present the performance (SR) of our model with varying numbers of in-context examples in Table 8. We observe a consistent increase in performance with the increase in the number of training examples. This suggests that while our zero/few-shot framework demonstrates strong performance in goal-oriented planning tasks, utilizing more annotated training data can lead to even better results.

## 6 Conclusions

We introduced VIDASSIST, a goal-oriented planning framework designed for challenging zero/few-shot learning settings in instructional videos. Our framework integrates LLMs and search-based techniques for superior long-horizon planning, achieving competitive or even better results than fully supervised approaches. In the future, we will extend our framework to even more complex tasks and video inputs that may last several hours. Furthermore, we will explore the design of more sophisticated search algorithms and value functions for improved zero-shot performance of our system.

**Acknowledgements.** We thank Yan-Bo Lin, Feng Cheng, Ce Zhang, Yue Yang, and Soumitri Chattopadhyay for their helpful discussions. This work was supported by the Sony Faculty Innovation Award, Laboratory for Analytic Sciences via NC State University, ONR Award N00014-23-1-2356.

## References

1. wikihow. <https://www.wikiHow.com/> 2
2. Abdelsalam, M.A., Rangrej, S.B., Hadji, I., Dvornik, N., Derpanis, K.G., Fazly, A.: Gepsan: Generative procedure step anticipation in cooking videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2988–2997 (2023) 6
3. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., et al.: Do as i can, not as i say: Grounding language in robotic affordances. arXiv preprint arXiv:2204.01691 (2022) 2, 5
4. Bellman, R.: A markovian decision process. *Journal of mathematics and mechanics* pp. 679–684 (1957) 2
5. Bi, J., Luo, J., Xu, C.: Procedure planning in instructional videos via contextual modeling and model-based policy learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15611–15620 (2021) 2, 3, 5, 10, 12
6. Chang, C.Y., Huang, D.A., Xu, D., Adeli, E., Fei-Fei, L., Niebles, J.C.: Procedure planning in instructional videos. In: European Conference on Computer Vision. pp. 334–350. Springer (2020) 2, 3, 5, 8, 9, 10, 11, 12
7. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., et al.: Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research* **24**(240), 1–113 (2023) 4
8. Finn, C., Yu, T., Zhang, T., Abbeel, P., Levine, S.: One-shot visual imitation learning via meta-learning. In: Conference on robot learning. pp. 357–368. PMLR (2017) 2, 4
9. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18995–19012 (2022) 10
10. Huang, W., Abbeel, P., Pathak, D., Mordatch, I.: Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In: International Conference on Machine Learning. pp. 9118–9147. PMLR (2022) 2, 4, 5, 6, 7, 9, 10, 11, 12
11. Islam, M.M., Ho, N., Yang, X., Nagarajan, T., Torresani, L., Bertasius, G.: Video recap: Recursive captioning of hour-long videos. arXiv preprint arXiv:2402.13250 (2024) 4, 5
12. Kahatapitiya, K., Ranasinghe, K., Park, J., Ryoo, M.S.: Language repository for long video understanding. arXiv preprint arXiv:2403.14622 (2024) 4
13. Kahneman, D.: Thinking, fast and slow. macmillan (2011) 2
14. Liu, J., Li, S., Wang, Z., Li, M., Ji, H.: A language-first approach for procedure planning. In: Findings of the Association for Computational Linguistics: ACL 2023. pp. 1941–1954 (2023) 2, 3, 5, 6, 8, 10, 11
15. Newell, A., Shaw, J.C., Simon, H.A.: Report on a general problem solving program. In: IFIP congress. vol. 256, p. 64. Pittsburgh, PA (1959) 6
16. Patel, D., Eghbalzadeh, H., Kamra, N., Iuzzolino, M.L., Jain, U., Desai, R.: Pre-trained language models as visual planners for human assistance. arXiv preprint arXiv:2304.09179 (2023) 1, 2, 3, 5, 8, 9, 10, 11, 14
17. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from



- natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) [8](#)
18. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019) [6](#), [7](#)
  19. Simon, H.A., Newell, A.: Human problem solving: The state of the theory in 1970. *American psychologist* **26**(2), 145 (1971) [6](#)
  20. Sloman, S.A.: The empirical case for two systems of reasoning. *Psychological bulletin* **119**(1), 3 (1996) [2](#)
  21. Song, C.H., Wu, J., Washington, C., Sadler, B.M., Chao, W.L., Su, Y.: Llm-planner: Few-shot grounded planning for embodied agents with large language models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2998–3009 (2023) [4](#)
  22. Sun, J., Huang, D.A., Lu, B., Liu, Y.H., Zhou, B., Garg, A.: Plate: Visually-grounded planning with transformers in procedural tasks. *IEEE Robotics and Automation Letters* **7**(2), 4924–4930 (2022) [2](#), [3](#), [5](#), [10](#), [12](#)
  23. Sünderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., Upcroft, B., Abbeel, P., Burgard, W., Milford, M., et al.: The limits and potentials of deep learning for robotics. *The International journal of robotics research* **37**(4-5), 405–420 (2018) [2](#), [4](#)
  24. Tang, Y., Ding, D., Rao, Y., Zheng, Y., Zhang, D., Zhao, L., Lu, J., Zhou, J.: Coin: A large-scale dataset for comprehensive instructional video analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1207–1216 (2019) [8](#), [10](#), [11](#), [13](#)
  25. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023) [5](#), [8](#), [9](#), [10](#), [11](#), [12](#)
  26. Wang, A.L., Lin, K.Y., Du, J.R., Meng, J., Zheng, W.S.: Event-guided procedure planning from instructional videos with text supervision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13565–13575 (2023) [2](#), [5](#), [10](#), [11](#), [12](#)
  27. Wang, H., Wu, Y., Guo, S., Wang, L.: Pdpp: Projected diffusion for procedure planning in instructional videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14836–14845 (2023) [11](#), [12](#)
  28. Xu, H., Ghosh, G., Huang, P.Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., Feichtenhofer, C.: Videoclip: Contrastive pre-training for zero-shot video-text understanding. arXiv preprint arXiv:2109.14084 (2021) [5](#), [8](#)
  29. Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., Narasimhan, K.: Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems* **36** (2024) [4](#), [6](#), [8](#)
  30. Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., Cao, Y.: React: Synergizing reasoning and acting in language models. arXiv preprint arXiv:2210.03629 (2022) [5](#), [6](#), [8](#)
  31. Zeng, A., Attarian, M., Ichter, B., Choromanski, K., Wong, A., Welker, S., Tombari, F., Purohit, A., Ryoo, M., Sindhwani, V., et al.: Socratic models: Composing zero-shot multimodal reasoning with language. arXiv preprint arXiv:2204.00598 (2022) [2](#), [4](#), [5](#)
  32. Zhang, C., Lu, T., Islam, M.M., Wang, Z., Yu, S., Bansal, M., Bertasius, G.: A simple llm framework for long-range video question-answering. arXiv preprint arXiv:2312.17235 (2023) [4](#)

- 33. Zhao, H., Hadji, I., Dvornik, N., Derpanis, K.G., Wildes, R.P., Jepson, A.D.: P3iv: Probabilistic procedure planning from instructional videos with weak supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2938–2948 (2022) 2, 3, 5, 6, 10, 11, 12
- 34. Zhao, Q., Zhang, C., Wang, S., Fu, C., Agarwal, N., Lee, K., Sun, C.: Antgpt: Can large language models help long-term action anticipation from videos? arXiv preprint arXiv:2307.16368 (2023) 2, 4
- 35. Zhukov, D., Alayrac, J.B., Cinbis, R.G., Fouhey, D., Laptev, I., Sivic, J.: Cross-task weakly supervised learning from instructional videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3537–3545 (2019) 8, 10, 11