# A Benchmark for Structured Procedural Knowledge Extraction from Cooking Videos

**Frank F. Xu**[1]* **Lei Ji**[2] **Botian Shi**[2] **Junyi Du**[3]
**Graham Neubig**[1] **Yonatan Bisk**[1] **Nan Duan**[2]
[1]Carnegie Mellon University [2]Microsoft Research [3]University of Southern California
{fangzhex,gneubig,ybisk}@cs.cmu.edu
{leiji,nanduan}@microsoft.com

## Abstract

Watching instructional videos are often used to learn about procedures. Video captioning is one way of automatically collecting such knowledge. However, it provides only an indirect, overall evaluation of multimodal models with no finer-grained quantitative measure of what they have learned. We propose instead, a benchmark of *structured* procedural knowledge extracted from cooking videos. This work is complementary to existing tasks, but requires models to produce interpretable structured knowledge in the form of verb-argument tuples. Our manually annotated open-vocabulary resource includes 356 instructional cooking videos and 15,523 video clip/sentence-level annotations. Our analysis shows that the proposed task is challenging and standard modeling approaches like unsupervised segmentation, semantic role labeling, and visual action detection perform poorly when forced to predict every action of a procedure in structured form.

## 1 Introduction

Instructional videos are a convenient way to learn a new skill. Although learning from video seems natural to humans, it requires identifying and understanding procedures and grounding them to the real world. In this paper, we propose a new task and dataset for extracting procedural knowledge into a fine-grained *structured* representation from *multimodal* information contained in a *large-scale* archive of *open-vocabulary* narrative videos with *noisy transcripts*. While there is a significant amount of related work (summarized in §3 & 7), to our knowledge there is no dataset similar in scope, with previous attempts focusing only on a single modality (e.g., text only (Kiddon et al., 2015) or video only (Zhukov et al., 2019; Alayrac et al., 2016)), using closed-domain taxonomies (Tang et al., 2019), or lacking structure in the procedural representation (Zhou et al., 2018a).

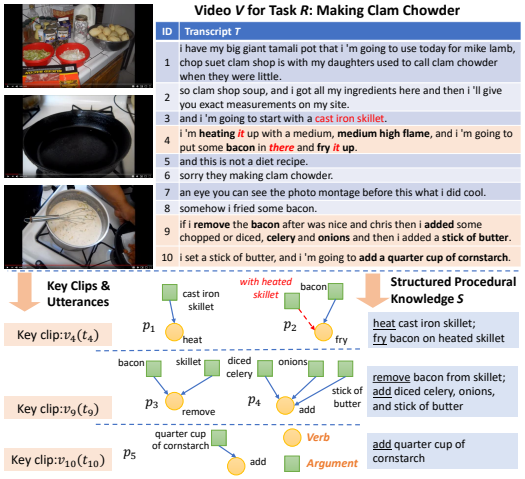


Figure 1: An example of extracting procedures for task *"Making Clam Chowder"*.

In our task, given a narrative video, say a cooking video on YouTube about *making clam chowder* as shown in Figure 1, our goal is to extract a series of tuples representing the procedure, e.g. (heat, cast iron skillet), (fry, bacon, with heated skillet), etc. We created a manually annotated, large test dataset for evaluation of the task, including over 350 instructional cooking videos along with over 15,000 English sentences in the transcripts spanning over 89 recipe types. This verb-argument structure using arbitrary textual phrases is motivated by open information extraction (Schmitz et al., 2012; Fader et al., 2011), but focuses on procedures rather than entity-entity relations.

This task is challenging with respect to both video and language understanding. For video, it requires understanding of video contents, with a spe-

* Work done at Microsoft Research Asia. Data and code: https://github.com/frankxu2004/cooking-procedural-extraction. Full version: https://arxiv.org/abs/2005.00706



*Proceedings of the First International Workshop on Natural Language Processing Beyond Text*, pages 30–40
Online, November 20, 2020. 
http://www.aclweb.org/anthology/W23-20%2d

cial focus on actions and procedures. For language, it requires understanding of oral narratives, including understanding of predicate-argument structure and coreference. In many cases it is necessary for both modalities to work together, such as when resolving null arguments necessitates the use of objects or actions detected from video contents in addition to transcripts. For example, the cooking video host may say "just a pinch of salt in", while adding some salt into a boiling pot of soup, in which case inferring the action "add" and its argument "pot" requires visual understanding.

Along with the novel task and dataset, we propose several baseline approaches that extract structure in a pipelined fashion. These methods first identify key clips/sentences using video and transcript information with unsupervised and supervised multimodal methods, then extract procedure tuples from the utterances and/or video of these key clips. On the utterances side, we utilize an existing state-of-the-art semantic role labeling model (Shi and Lin, 2019), with the intuition that semantic role labeling captures the verb-argument structures of a sentence, which would be directly related to procedures and actions. On the video side, similarly, we utilize existing state-of-the-art video action/object recognition model trained in kitchen settings to further augment utterance-only extraction results. The results are far from perfect, demonstrating that the proposed task is challenging and that structuring procedures requires more than just state-of-the-art semantic parsing or video action recognition.

## 2 Problem Definition

We show a concrete example of our procedural knowledge extraction task in Figure 1. Our ultimate goal is to automatically map *unstructured* instructional video (clip and utterances) to *structured* procedures, defining what actions should be performed on which objects, with what arguments and in what order. We define the input to such an extraction system:

- Task $R$, e.g. "Create Chicken Parmesan" and instructional video $V_R$ describing the procedure to achieve task $R$, e.g. a video titled "Chicken Parmesan - Let's Cook with ModernMom".[1]
- A sequence of $n$ sentences $T_R = \{t_0, t_1, ..., t_n\}$ representing video $V_R$'s corresponding transcript. According to the time stamps of the transcript sentences, the video is also segmented into $n$ clips $V_R = \{v_0, v_1, ..., v_n\}$ accordingly to align with the sentences in the transcript $T_R$.

| | Ours | AR | YC2 | CT | COIN | How2 | HAKE | TACOS |
|---|---|---|---|---|---|---|---|---|
| General domain? | | | | ✓ | ✓ | ✓ | ✓ | |
| Multimodal input? | ✓ | | | | | ✓ | ✓ | ✓ |
| Use transcript? | ✓ | | | | | ✓ | | |
| Use noisy text? | ✓ | | | | | ✓ | | |
| Open extraction? | ✓ | | ✓ | | | | | |
| Structured format? | ✓ | ✓ | | ✓ | | | ✓ | ✓ |

Table 1: Comparison to current datasets.

The output will be:

- A sequence of $m$ procedure tuples $S_R = \{s_0, s_1, ..., s_m\}$ describing the key steps to achieve task $R$ according to instructional video $V_R$.
- An identified list of *key* video clips and corresponding sentences $V'_R \subseteq V_R$, to which procedures in $S_R$ are grounded.

Each procedural tuple $s_j = (\text{verb}, \text{arg}_1, ..., \text{arg}_k) \in S_R$ consists of a verb phrase and its arguments. Only the "verb" field is required, and thus the tuple size ranges from 1 to $k+1$. All fields can be either a word or a phrase.

Not every clip/sentence describes procedures, as most videos include an intro, an outro, non-procedural narration, or off-topic chit-chat. Key clips $V'_R$ are clips associated with one or more procedures in $P_R$, with some clips/sentences associated with multiple procedure tuples. Conversely, each procedure tuple will be associated with only a single clip/sentence.

## 3 Dataset & Analysis

While others have created related datasets, they fall short on key dimensions which we remedy in our work. Specifically, In Table 1 we compare to AllRecipes (Kiddon et al., 2015) (AR), YouCook2 (Zhou et al., 2018b) (YC2), CrossTask (Zhukov et al., 2019) (CT), COIN (Tang et al., 2019), How2 (Sanabria et al., 2018), HAKE (Li et al., 2019) and TACOS (Regneri et al., 2013). Additional details about datasets are included in the Appendix A.[2] In summary, none have both *structured* and *open* extraction annotations for the procedural knowledge extraction task, since most focus on either video summarization/captioning or action localization/classification.

[1] https://www.youtube.com/watch?v=nWGpCmDlNU4

[2] A common dataset we do not include here is HowTo100M (Miech et al., 2019) as it does not contain any annotations.


Figure 2: Annotation interface.

| | Verbs | Arguments |
|---|---|---|
| Total # | 4004 | 6070 |
| Average # per key clip | 1.12 | 1.70 |
| Average #words | 1.07 | 1.43 |
| % directly from transcript | 69.8 | 75.0 |
| % coreference (pronouns) | N/A | 14.4 |
| % ellipsis | 30.2 | 10.6 |

Table 2: Statistics of annotated verbs and arguments in procedures.

### 3.1 Dataset Creation

To address the limitations of existing datasets, we created our own evaluation dataset by annotating structured procedure knowledge given the video and transcript. Native English-speakers annotated four videos per recipe type (e.g. clam chowder, pizza margherita, etc.) in the YouCook2 dataset into the structured form presented in §2 (totaling 356 videos). Annotators selected key clips as important steps and extracted corresponding fields to fill in verbs and arguments. Filling in the fields with the original tokens was preferred but not required (e.g., in cases of coreference and ellipsis). The result is a series of video clips labeled with procedural structured knowledge as a sequence of steps $s_j$ and series of short sentences describing the procedure.

Figure 2 shows the user interface of annotation tool. The process is divided into 3 questions per clip: **Q1:** Determine if the video clip is a key step if: (1) the clip or transcript contains at least one action; (2) the action is required for accomplishing the task (i.e. not a self introduction); and (3) for if a clip duplicates a previous key clip, choose the one with clearer visual and textual signals (e.g. without coreference, etc.). **Q2:** For each key video clip, annotate the key procedural tuples. We have

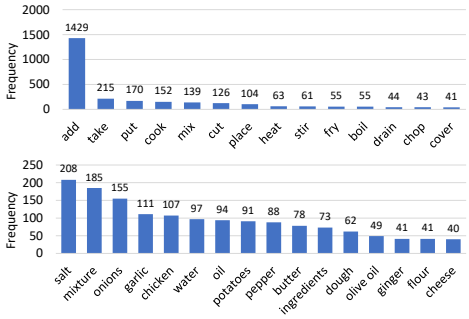


Figure 3: Most frequent verbs (upper) and arguments (lower).

annotators indicate which actions are both seen and mentioned by the instructor in the video. The actions should correspond to a verb and its arguments from the original transcript except in the case of ellipsis or coreference where they have to refer to earlier phrases based on the visual scene. **Q3:** Construct a short fluent sentence from the annotated tuples for the given video clip.

We have two expert annotators and a professional labeling supervisor for quality control and deciding the final annotations. To improve the data quality, the supervisor reviewed all labeling results, and applied several heuristic rules to find anomalous records for further correction. The heuristic is to check the annotated verb/arguments that are not found in corresponding transcript text. Among these anomalies, the supervisor checks the conflicts between the two annotators. 25% of all annotations were modified as a result. On average annotators completed task Q1 at 240 sentences (clips) per hour and task Q2 and Q3 combined at 40 sentences per hour. For Q1, we observe an inter-annotator agreement with Cohen's Kappa of 0.83.[3] Examples are shown in Table 3.

### 3.2 Dataset Analysis

Overall, the dataset contains 356 videos with 15,523 video clips/sentences, among which 3,569 clips are labeled as key steps. Sentences average 16.3 tokens, and the language style is oral English. For structured procedural annotations, there are 347 unique verbs and 1,237 unique objects in all. Statistics are shown in Table 2. Figure 3 lists the most commonly appearing verbs and entities. The action *add* is most frequently performed, and the entities *salt* and *onions* are the most popular ingredients.

[3] We use the Jaccard ratio between the annotated tokens of two annotators for Q2's agreement. Verb annotations have a higher agreement at 0.77 than that of arguments at 0.72.

| Transcript sentence | Procedure summary | Verb | Arguments | |
|---|---|---|---|---|
| so we've **placed the dough directly into the caputo flour** that we import from italy. | place dough in caputo flour | place | dough | caputo flour |
| we just **give *(ellipsis)* a squish with our palm and make *it* flat in the center**. | squish dough with palm | squish | dough | with palm |
| | flatten center of dough | flatten | center of dough | |
| so will have to **rotate *it* every thirty to forty five seconds** ... | rotate pizza every 30-45 seconds | rotate | pizza | every 30-45 seconds |

Table 3: Annotations of structured procedures and summaries. *Coreference* and *ellipsis* are marked with *italics* and are resolved into referred phrases also linked back in the annotations. See Appendix (Table 6) for more examples.

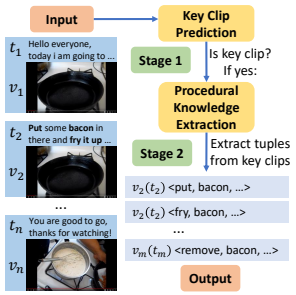


Figure 4: Extraction pipeline.

In nearly 30% of annotations, some verbs and arguments cannot be directly found in the transcript. An example is "(add) some salt into the pot", and we refer to this variety of absence as *ellipsis*. Arguments not mentioned explicitly are mainly due to (1) pronoun references, e.g. "put it (fish) in the pan"; (2) ellipsis, where the arguments are absent from the oral language, e.g. "put the mixture inside" where the argument "oven" is omitted. The details can be found in Table 2. The coreferences and ellipsis phenomena add difficulty to our task, and indicate the utility of using multimodal information from the video signal and contextual procedural knowledge for inference.

## 4 Extraction Stage 1: Key Clip Selection

In this and the following section, we describe our two-step pipeline for procedural knowledge extraction (also in Figure 4). This section describes the first stage of determining which clips are "key clips" that contribute to the description of the procedure. We describe several key clip selection models, which consume the transcript and/or the video within the clip and decide whether it is a key clip.

### 4.1 Parsing-Based Heuristic Baselines

Given our unsupervised setting, we first examine two heuristic parsing-based methods that focus on the transcript only, one based on semantic role labeling (SRL) and the other based on an unsupervised segmentation model Kiddon et al. (2015).

Before introducing heuristic baselines, we note that having a lexicon of domain-specific actions will be useful, e.g., for filtering pretrained model outputs, or providing priors to the unsupervised model described later. In our cooking domain, these actions can be expected to consist mostly of verbs related to cooking actions and procedures. Observing recipe datasets such as AllRecipes (Kiddon et al., 2015) or WikiHow (Miech et al., 2019; Zhukov et al., 2019), we find that they usually use imperative and concise sentences for procedures and the first word is usually the action verb like *"add"*, e.g., *add some salt into the pot*. We thus construct a cooking lexicon by aggregating the frequently appearing verbs as the first word from AllRecipes, with frequency over a threshold of 5. We further filter out words that have no verb synsets in WordNet (Miller, 1995). Finally we manually filter out noisy or too general verbs like "go". Note that when applying to other domains, the lexicon can be built following a similar process of first finding a domain-specific corpus with simple and formal instructions, and then obtaining the lexicon by aggregation and filtering.

**Semantic role labeling baselines.** One intuitive trigger in the transcript for deciding whether the sentence is a key step should be the action words, i.e. the verbs. In order to identify these action words we use semantic role labeling (Gildea and Jurafsky, 2002), which analyzes natural language sentences to extract information about "who did what to whom, when, where and how?" The output is in the form of predicates and their respective arguments that acts as semantic roles, where the verb acts as the root (head) of the parse. We run a strong semantic role labeling model (Shi and Lin, 2019) included in the AllenNLP toolkit (Gardner et al., 2018) on each sentence in the transcript. From the output we get a set of verbs for each of the sentences.[4] Because not all verbs in all sentences represent actual key actions for the procedure, we

[4]The SRL model is used in this stage only as a verb identifier, with other output information used in stage 2.

additionally filter the verbs with the heuristically created cooking lexicon above, counting a clip as a key clip only if at least one of the SRL-detected verbs is included in the lexicon.

**Unsupervised recipe segmentation baseline (Kiddon et al., 2015).** The second baseline is based on the outputs of the unsupervised recipe sentence segmentation model in Kiddon et al. (2015). Briefly speaking, the model is a generative probabilistic model where verbs and arguments, together with their numbers, are modeled as latent variables. It uses a bigram model for string selection. It is trained on the whole transcript corpus of YouCook2 videos iteratively for 15 epochs using a hard EM approach before the performance starts to converge. The count of verbs in the lexicon created in §4.1 is provided as a prior through initialization. We then do inference to parse the transcripts in our dataset using the trained model. Following the same heuristics as the SRL outputs, we treat sentences with non-empty parsed predicates after lexical filtering as key sentences, and those without as negatives.

### 4.2 Neural Selection Baseline

Next, we implement a supervised neural network model that incorporates visual information, which we have posited before may be useful in the face of incomplete verbal utterances. We extract the features of the sentence and each video frame using pretrained feature extractors respectively. Then we perform attention (Bahdanau et al., 2014) over each frame feature, using the sentence as a query, in order to acquire the representation of the video clip. Finally, we combine the visual and textual features to predict whether the input is a key clip. The model is trained on a *general domain* instructional key clip selection dataset with *no* overlap with ours, and our annotated dataset is used for evaluation *only*. Additional details about the model and training dataset are included in Appendix B.

## 5 Extraction Stage 2: Structured Knowledge Extraction

With the identified key clips and corresponding transcript sentences, we proceed to the second stage that performs clip/sentence-level procedural knowledge extraction from key clips. In this stage, the extraction is done from clips that are identified at first as "key clips".

### 5.1 Extraction From Utterances

We first present two baselines to extract structured procedures using transcripts only, similarly to the key-clip identification methods described in §4.1.

**Semantic role labeling.** For the first baseline, we use the same pretrained SRL model introduced in §4.1 to conduct inference on the sentences in key clips identified from stage 1. Because they consist of verb-argument structures, the outputs of the SRL model are well aligned with the task of extracting procedural tuples that identify actions and their arguments. However, not all outputs from the SRL model are the structured procedural knowledge we aim to extract. For example, in the sentence *"you 're ready to add a variety of bell peppers"* from the transcript, the outputs from SRL model contains two parses with two predicates, *"are"* and *"add"*, where only the latter is actually part of the procedure. To deal with this issue we first perform filtering similar to that used in stage 1, removing parses with predicates (verbs) outside of the domain-specific action lexicon we created in §4.1. Next, we filter out irrelevant arguments in the parse. For example, the parse from the SRL model for sentence "I add a lot of pepper because I love it." after filtering out irrelevant verb "love" is "[ARG0: I] [V: add] [ARG1: a lot of pepper] [ARGM-CAU: because I love it]", some arguments such as ARG0 and ARGM-CAU are clearly not contributing to the procedure. We provide a complete list of the filtered argument types in Appendix C.

**Unsupervised recipe segmentation (Kiddon et al., 2015).** The second baseline is to use the same trained segmentation model as in §4.1 to segment selected key transcript sentences into verbs and arguments. We treat segmented predicates in the key sentence as procedural verbs, and segmented predicate arguments plus preposition arguments as procedural arguments.

### 5.2 Extraction From Video

We also examine a baseline that utilizes two forms of visual information in videos: actions and objects. We predict both verbs and nouns of a given video clip via a state-of-the-art action detection model TSM (Lin et al., 2019),[5] trained on the EpicKitchen (Damen et al., 2018a) dataset.[6] For each video, we extract 5-sec video segments and feed

[5] https://github.com/epic-kitchens/action-models

[6] https://epic-kitchens.github.io/2019

| | Acc | P | R | F1 |
|---|---|---|---|---|
| Parsing-based Heuristics | | | | |
| SRL w/o heur. | 25.9 | 23.4 | **97.6** | 37.7 |
| SRL w/ heur. | 61.2 | 35.2 | 81.4 | 49.1 |
| Kiddon et al. (2015) | 67.3 | 33.5 | 42.7 | 37.6 |
| Neural Model | | | | |
| Visual Only | 43.8 | 27.2 | 85.9 | 41.3 |
| Text Only | 76.3 | 49.0 | 78.1 | 60.2 |
| V+T (Full Model) | **77.7** | **51.0** | 75.3 | **60.8** |

Table 4: Key clip selection results.

into the action detection model. The outputs of the models are in a *predefined* set of labels of verbs (actions) and nouns (objects).[7] We directly combine the outputs from the model on each video segment, aggregate and temporally align them with key clips/sentences, forming the final output.

### 5.3 Utterance and Video Fusion

Finally, to take advantage of the fact that utterance and video provide complementary views, we perform multimodal fusion of the results of both of these model varieties. We adopt a simple method of fusion by taking the union of the verbs/actions and arguments/objects respectively from the best performing utterance-only model and the visual detection model.

## 6 Evaluation

We propose evaluation metrics and provide evaluation results on our annotated dataset for both of the two stages: key clip selection and structured procedural extraction. Detailed reproducibility information about the experiments are in Appendix F. Besides quantitative evaluation and qualitative evaluations, we also analyze the key challenges of this task.

### 6.1 Extraction Stage 1: Key Clip Selection

In this section, we evaluate results of the key clip selection described in §4. We evaluate using the accuracy, precision, recall and F1 score for the binary classification problem of whether a given clip in the video is a key clip. The results are shown in Table 4. We compare parsing-based heuristic models and supervised neural models, with ablations (model details in Appendix B). From the experimental results in Table 4, we can see that:

1. Unsupervised heuristic methods perform worse than neural models with training data. This is despite the fact that the dataset used for training neural models has a different data distribution and domain from the test set.
2. Among heuristic methods, pretrained SRL is better than Kiddon et al. (2015) even though the second is trained on transcript text from YouCook2 videos. One possible reason is that the unsupervised segmentation method was specially designed for recipe texts, which are mostly simple, concise and imperative sentences found in recipe books, while the transcript is full of noise and tends to have longer, more complicated, and oral-style English.
3. Post-processing significantly improves the SRL model, showing that filtering unrelated arguments and incorporating the cooking lexicon helps, especially with reducing false positives.
4. Among neural method ablations, the model using only visual features performs worse than that using only text features. The best model for identifying key clips among proposed baselines uses both visual and text information in the neural model.

Besides quantitative evaluation, we analyzed key clip identification results and found a number of observations. First, background introductions, advertisements for the YouTube channel, etc. can be relatively well classified due to major differences both visually and textually from procedural clips. Second, alignment and grounding between the visual and textual domains is crucial for key clip prediction, yet challenging. For example, the clip with the transcript sentence "add more pepper according to your liking" is identified as a key clip. However, it is in fact merely a suggestion made by the speaker about an imaginary scenario, rather than a real action performed and thus should not be regarded as a key procedure.

### 6.2 Extraction Stage 2: Structured Procedure Extraction

In this stage, we perform key clip-level evaluation for structured procedural knowledge extraction by matching the ground truth and predicted structures with both exact match and two fuzzy scoring strategies. To better show how stage 1 performance affects the whole pipeline, we evaluate on both ground truth (oracle) and predicted key clips. Similarly to the evaluation of key clip selection, we compare the parsing-based methods (§5.1), as well as purposing the action detection results from

[7] Notably, this contrasts to our setting of attempting to recognize into an open label set, which upper-bounds the accuracy of any model with a limited label set.

| Model | Verbs | | | | | | | | | Arguments | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Exact Match | | | Fuzzy | | | Partial Fuzzy | | | Exact Match | | | Fuzzy | | | Partial Fuzzy | | |
| | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 |
| Using oracle key clips | | | | | | | | | | | | | | | | | | |
| Kiddon et al. (2015) | 12.0 | 10.9 | 11.4 | 18.8 | 17.2 | 18.0 | 20.2 | 18.4 | 19.3 | 0.4 | 0.9 | 0.5 | 10.4 | 19.3 | 13.5 | 16.4 | 30.2 | 21.3 |
| SRL w/o heur. | 19.4 | 54.7 | 28.6 | 25.3 | 70.1 | 37.2 | 26.6 | 73.8 | 39.1 | 1.3 | **5.4** | 2.0 | 14.1 | **53.6** | 22.3 | 22.0 | **81.8** | 34.6 |
| SRL w/ heur. | **38.7** | 51.6 | **44.3** | **45.2** | 60.3 | **51.7** | **46.9** | 62.6 | **53.6** | **1.6** | 3.3 | **2.2** | **21.2** | 39.8 | **27.7** | **32.3** | 59.5 | **41.9** |
| Visual | 4.1 | 6.7 | 5.1 | 17.9 | 27.8 | 21.7 | 19.3 | 30.1 | 23.5 | 0.9 | 1.1 | 1.0 | 17.8 | 25.8 | 21.1 | 24.2 | 36.2 | 29.0 |
| Fusion | 19.9 | **55.2** | 29.3 | 28.6 | **73.3** | 41.2 | 31.2 | **78.6** | 44.7 | 1.1 | 3.8 | 1.6 | 16.9 | 50.0 | 25.2 | 24.4 | 72.5 | 36.5 |
| Using predicted key clips | | | | | | | | | | | | | | | | | | |
| Kiddon et al. (2015) | 7.0 | 6.3 | 6.6 | 10.9 | 10.0 | 10.4 | 11.7 | 10.7 | 11.2 | 0.2 | 0.5 | 0.3 | 6.1 | 11.2 | 7.9 | 9.5 | 17.5 | 12.3 |
| SRL w/o heur. | 11.2 | 31.7 | 16.6 | 14.7 | 40.7 | 21.6 | 15.4 | 42.8 | 22.6 | 0.7 | **3.1** | 1.2 | 8.2 | **31.1** | 13.0 | 12.7 | **47.4** | 20.1 |
| SRL w/ heur. | **22.5** | 29.9 | **25.7** | **26.2** | 35.0 | **30.0** | **27.2** | 36.3 | **31.1** | **0.9** | 1.9 | **1.3** | **12.3** | 23.1 | **16.1** | **18.8** | 34.5 | **24.3** |
| Visual | 2.4 | 3.9 | 3.0 | 10.4 | 16.1 | 12.6 | 11.2 | 17.5 | 13.7 | 0.5 | 0.6 | 0.6 | 10.3 | 15.0 | 12.2 | 14.1 | 21.0 | 16.8 |
| Fusion | 11.5 | **32.0** | 17.0 | 16.6 | **42.5** | 23.9 | 18.1 | **45.6** | 25.9 | 0.6 | 2.2 | 1.0 | 9.8 | 29.0 | 14.6 | 14.1 | 42.1 | 21.2 |

Table 5: Clip/sentence-level structured procedure extraction results for verbs and arguments.

video signals for our task. Besides, we compare utterance-only and video-only baselines with our naive multi-modal fusion method.

We evaluate with respect to precision, recall and the F1 measure. Similarly to the evaluation method used for SRL (Carreras and Màrquez, 2004), precision (P) is the proportion of verbs or arguments predicted by a model which are correct, i.e. $TP/\#$predicted where $TP$ is the number of true positives. Recall (R) is the proportion of correct verbs or arguments which are predicted by a model, i.e. $TP/\#$gold. The key here is how to calculate $TP$ and we propose 3 methods: exact match, fuzzy matching, and partial fuzzy matching. The first is straight forward, we count true positives if and only if the predicted phrase is an exact string match in the gold phrases. However, because our task lies in the realm of open phrase extraction without predefined labels, it is unfairly strict to count only the exact string matches as $TP$. Also by design, the gold extraction results cannot always be found in the original transcript sentence (refer to §3.2), so we are also unable to use token-based metrics as in sequence tagging (Sang and De Meulder, 2003), or span-based metrics as in some question answering tasks (Rajpurkar et al., 2016). Thus for the second metric we call *"fuzzy"*, we leverage edit distance to enable fuzzy matching and assign a "soft" score for $TP$. In some cases, the two strings of quite different lengths will hurt the *fuzzy* score due to the nature of edit distance, even though one string is a substring of another. To get around this, we propose a third metric, *"partial fuzzy"* to get the score of the best matching substring with the length of the shorter string in comparison. Note that this third metric will bias towards shorter, correct phrases and thus we should have a holistic view of all 3 metrics during the evaluation. Details of two fuzzy metrics are described in Appendix D.

Table 5 illustrates evaluation results:

1. Argument extraction is much more challenging compared to verb extraction, according the results: arguments contain more complex types of phrases (e.g. objects, location, time, etc.) and are longer in length. It is hard to identify complex arguments with our current heuristic or unsupervised baselines and thus the need for better supervised or semi-supervised models.
2. Heuristic SRL methods perform better than the unsupervised segmentation model even though the second is trained on our corpus. This demonstrates the generality of SRL models, but the heuristics applied at the output of SRL models still improve the performance by reducing false positives.
3. The visual-only method performs the worst, mainly because of the domain gap between visual detection model outputs and our annotated verbs and arguments. Other reasons include: the closed label set predefined in EpicKitchen; challenges in domain transferring from closed to open extraction; different video data distribution between EpicKitchen (for training) and our dataset (YouCook2, for testing); limited performance of video detection model itself.
4. Naive multimodal fusion leads to an overall performance drop to below the utterance-only model, partly due to the differences in video data distribution and domain, as well as the limitation of the predefined set of verbs and nouns in the EpicKitchen dataset, implying the need for better multimodal fusion method. Unsurprisingly, the recall for verb extraction raises after the fusion, suggesting that action detection in videos helps with the coverage. The drop in argument extraction suggests the complexity of arguments in our open extraction setting: it

should be more than mere object detection.

Besides quantitative results, we also showcase qualitative analysis of example extraction outputs in Appendix E. From both, we suggest that there are two key challenges moving forward:

**Verb extraction:** We find that verb ellipsis is common in transcripts. The transcript text contains sentences where key action "verbs" do not have verb part-of-speech in the sentence. For example, in the sentence "give it a flip ..." with the annotation ("flip", "pancake"), the model detects "give" as the verb rather than "flip". Currently all our baselines are highly reliant on a curated lexicon for verb selection and thus such cases will get filtered out. How to deal with such cases with general verbs like *make*, *give*, *do* remains challenging and requires extracting from the contexts.

**Argument extraction:** Speech-to-text errors are intrinsic in automatically acquired transcripts and cause problems during parsing that cascade. Examples are that "add flour" being recognized as "add flower" and "sriracha sauce" being recognized as "sarrah cha sauce" causing wrong extraction outputs. Coreference and ellipsis are also challenging and hurting current benchmark performance, as our baselines do not tackle any of these explicitly. Visual co-reference and language grounding (Huang et al., 2018, 2017) provides a feasible method for us to tackle these cases in the future.

## 7 Related Work

**Text-based procedural knowledge extraction.** Procedural text understanding and knowledge extraction (Chu et al., 2017; Park and Motahari Nezhad, 2018; Kiddon et al., 2015; Jermsurawong and Habash, 2015; Liu et al., 2016; Long et al., 2016; Maeta et al., 2015; Malmaud et al., 2014; Artzi and Zettlemoyer, 2013; Kuehne et al., 2017) has been studied for years on step-wise textual data such as WikiHow. Chu et al. (2017) extracted open-domain knowledge from how-to communities. Recently Zhukov et al. (2019) also studied to adopt the well-written how-to data as weak supervision for instructional video understanding. Unlike existing work on action graph/dependency extraction (Kiddon et al., 2015; Jermsurawong and Habash, 2015), our approach differs as we extract knowledge from the visual signals and transcripts directly, not from imperative recipe texts.

**Instructional video understanding.** Beyond image semantics (Yatskar et al., 2016), unlike existing tasks for learning from instructional video (Zhou et al., 2018c; Tang et al., 2019; Alayrac et al., 2016; Song et al., 2015; Sener et al., 2015; Huang et al., 2016; Sun et al., 2019b,a; Plummer et al., 2017; Palaskar et al., 2019), combining video & text information in procedures (Yagcioglu et al., 2018; Fried et al., 2020), visual-linguistic reference resolution (Huang et al., 2018, 2017), visual planning (Chang et al., 2019), joint learning of object and actions (Zhukov et al., 2019; Richard et al., 2018; Gao et al., 2017; Damen et al., 2018b), pretraining joint embedding of high level sentence with video clips (Sun et al., 2019b; Miech et al., 2019), our task proposal requires explicit structured knowledge tuple extraction.

In addition to closely related work (§3) there is a wide literature (Malmaud et al., 2015; Zhou et al., 2018b; Ushiku et al., 2017; Nishimura et al., 2019; Tang et al., 2019; Huang et al., 2016; Shi et al., 2019; Ushiku et al., 2017) that aims to predict/align dense procedural captions given the video, which are the most similar works to ours. Zhou et al. (2018c) extracted temporal procedures and then generated captioning for each procedure. Sanabria et al. (2018) proposes a multimodal abstractive summarization for how-to videos with either human labeled or speech-to-text transcript. Alayrac et al. (2016) also introduces an unsupervised step learning method from instructional videos. Inspired by cross-task sharing (Zhukov et al., 2019), which is a weakly supervised method to learn shared actions between tasks, fine grained action and entity are important for sharing similar knowledge between various tasks. We focus on *structured* knowledge of fine-grained actions and entities.Visual-linguistic coreference resolution (Huang et al., 2018, 2017) is among one of the open challenges for our proposed task.

## 8 Conclusions & Open Challenges

We propose a multimodal open procedural knowledge extraction task, present a new evaluation dataset, produce benchmarks with various methods, and analyze the difficulties in the task. Meanwhile we investigate the limit of existing methods and many open challenges for procedural knowledge acquisition, including: to better deal with cases of coreference and ellipsis in visual-grounded languages; exploit cross-modalities of information with more robust, semi/un-supervised models; potential improvement from structured knowledge in downstream tasks (e.g., video captioning).

## References


Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. 2016. Unsupervised learning from narrated instruction videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4575–4583.

Yoav Artzi and Luke Zettlemoyer. 2013. Weakly supervised learning of semantic parsers for mapping instructions to actions. *Transactions of the Association for Computational Linguistics*, 1:49–62.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Xavier Carreras and Lluís Màrquez. 2004. Introduction to the conll-2004 shared task: Semantic role labeling. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 89–97.

Chien-Yi Chang, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. 2019. Procedure planning in instructional videos. *ArXiv*, abs/1907.01172.

Cuong Xuan Chu, Niket Tandon, and Gerhard Weikum. 2017. Distilling task knowledge from how-to communities. In *Proceedings of the 26th International Conference on World Wide Web*, pages 805–814. International World Wide Web Conferences Steering Committee.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2018a. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. 2018b. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1535–1545. Association for Computational Linguistics.

Daniel Fried, Jean-Baptiste Alayrac, Phil Blunsom, Chris Dyer, Stephen Clark, and Aida Nematzadeh. 2020. Learning to segment actions from observation and narration.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5267–5275.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. *arXiv preprint arXiv:1803.07640*.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational linguistics*, 28(3):245–288.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

De-An Huang, Shyamal Buch, Lucio Dery, Animesh Garg, Li Fei-Fei, and Juan Carlos Niebles. 2018. Finding "it": Weakly-supervised, reference-aware visual grounding in instructional videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. 2016. Connectionist temporal modeling for weakly supervised action labeling. In *European Conference on Computer Vision*, pages 137–153. Springer.

De-An Huang, Joseph J Lim, Li Fei-Fei, and Juan Carlos Niebles. 2017. Unsupervised visual-linguistic reference resolution in instructional videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2183–2192.

Jermsak Jermsurawong and Nizar Habash. 2015. Predicting the structure of cooking recipes. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 781–786.

Chloé Kiddon, Ganesa Thandavam Ponnuraj, Luke S. Zettlemoyer, and Yejin Choi. 2015. Mise en place: Unsupervised interpretation of instructional recipes. In *EMNLP*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Hilde Kuehne, Alexander Richard, and Juergen Gall. 2017. Weakly supervised learning of actions from transcripts. *Computer Vision and Image Understanding*, 163:78–89.

Yong-Lu Li, Liang Xu, Xijie Huang, Xinpeng Liu, Ze Ma, Mingyang Chen, Shiyi Wang, Hao-Shu Fang, and Cewu Lu. 2019. Hake: Human activity knowledge engine. *arXiv preprint arXiv:1904.06539*.

Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE International Conference on Computer Vision*.

Changsong Liu, Shaohua Yang, Sari Saba-Sadiya, Nishant Shukla, Yunzhong He, Song-Chun Zhu, and Joyce Chai. 2016. Jointly learning grounded task structures from language instruction and visual demonstration. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1482–1492.

Reginald Long, Panupong Pasupat, and Percy Liang. 2016. Simpler context-dependent logical forms via model projections. *arXiv preprint arXiv:1606.05378*.

Hirokuni Maeta, Tetsuro Sasada, and Shinsuke Mori. 2015. A framework for procedural text understanding. In *Proceedings of the 14th International Conference on Parsing Technologies*, pages 50–60.

Jonathan Malmaud, Jonathan Huang, Vivek Rathod, Nicholas Johnston, Andrew Rabinovich, and Kevin Murphy. 2015. What's cookin'? interpreting cooking videos using text, speech and vision. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 143–152, Denver, Colorado. Association for Computational Linguistics.

Jonathan Malmaud, Earl Wagner, Nancy Chang, and Kevin Murphy. 2014. Cooking with semantics. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 33–38.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. *arXiv:1906.03327*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

James Munkres. 1957. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38.

Taichi Nishimura, Atsushi Hashimoto, Yoko Yamakata, and Shinsuke Mori. 2019. Frame selection for producing recipe with pictures from an execution video of a recipe. In *Proceedings of the 11th Workshop on Multimedia for Cooking and Eating Activities*, pages 9–16. ACM.

Shruti Palaskar, Jindrich Libovickỳ, Spandana Gella, and Florian Metze. 2019. Multimodal abstractive summarization for how2 videos. *arXiv preprint arXiv:1906.07901*.

Hogun Park and Hamid Reza Motahari Nezhad. 2018. Learning procedures from text: Codifying how-to procedures in deep neural networks. In *Companion Proceedings of the The Web Conference 2018*, pages 351–358. International World Wide Web Conferences Steering Committee.

Bryan A Plummer, Matthew Brown, and Svetlana Lazebnik. 2017. Enhancing video summarization via vision-language embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5781–5789.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36.

Alexander Richard, Hilde Kuehne, and Juergen Gall. 2018. Action sets: Weakly supervised action segmentation without ordering constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5987–5996.

Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. 2018. How2: a large-scale dataset for multimodal language understanding. *arXiv preprint arXiv:1811.00347*.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics.

Ozan Sener, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. 2015. Unsupervised semantic parsing of video collections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4480–4488.

Botian Shi, Lei Ji, Yaobo Liang, Nan Duan, Peng Chen, Zhendong Niu, and Ming Zhou. 2019. Dense procedure captioning in narrated instructional videos. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 6382–6391.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*.

Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. 2015. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187.

Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. 2019a. Learning video representations using contrastive bidirectional transformer.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019b. Videobert: A joint model for video and language representation learning. *arXiv preprint arXiv:1904.01766*.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. 2019. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1207–1216.

Atsushi Ushiku, Hayato Hashimoto, Atsushi Hashimoto, and Shinsuke Mori. 2017. Procedural text generation from an execution video. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 326–335, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Semih Yagcioglu, Aykut Erdem, Erkut Erdem, and Nazli Ikizler-Cinbis. 2018. RecipeQA: A challenge dataset for multimodal comprehension of cooking recipes. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1358–1368, Brussels, Belgium. Association for Computational Linguistics.

Mark Yatskar, Luke Zettlemoyer, and Ali Farhadi. 2016. Situation recognition: Visual semantic role labeling for image understanding. In *Conference on Computer Vision and Pattern Recognition*.

Luowei Zhou, Nathan Louis, and Jason J Corso. 2018a. Weakly-supervised video object grounding from text by loss weighting and object interaction. *arXiv preprint arXiv:1805.02834*.

Luowei Zhou, Chenliang Xu, and Jason J Corso. 2018b. Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. 2018c. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748.

Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. 2019. Cross-task weakly supervised learning from instructional videos. In *Computer Vision and Pattern Recognition (CVPR)*.