Eliciting Affective Events from Language Models by Multiple View Co-prompting

Yuan Zhuang and **Ellen Riloff**
Kahlert School of Computing
University of Utah
Salt Lake City, UT 84112
{yyzhuang, riloff}@cs.utah.edu

Abstract

Prior research on affective event classification showed that exploiting weakly labeled data for training can improve model performance. In this work, we propose a simpler and more effective approach for generating training data by automatically acquiring and labeling affective events with *Multiple View Co-prompting*, which leverages two language model prompts that provide independent views of an event. The approach starts with a modest amount of gold data and prompts pre-trained language models to generate new events. Next, information about the probable affective polarity of each event is collected from two complementary language model prompts and jointly used to assign polarity labels. Experimental results on two datasets show that the newly acquired events improve a state-of-the-art affective event classifier. We also present analyses which show that using multiple views produces polarity labels of higher quality than either view on its own.

1 Introduction

People's emotional states are influenced by the events that they experience. For example, people typically feel happy when they graduate with a degree or get a new job, but become upset when they get fired or lose personal property. Prior work (Ding and Riloff, 2016, 2018) has referred to events that positively or negatively impact people as *affective events*. In this work, we study the task of affective event classification, which determines whether the polarity of a given event is positive, negative or neutral. For example, *"I graduated from college"* would be Positive, but *"I broke my leg"* would be Negative.

Previous research has shown that the performance of affective event classification models is limited by the amount of gold training data (Zhuang et al., 2020), which is costly to annotate and not readily available in large quantities. Recently, researchers have been developing methods to generate more training data by extracting events from text corpora and assigning polarity labels with weakly supervised methods (Saito et al., 2019; Zhuang et al., 2020). However, these methods pose practical challenges, including in some cases the need to acquire data from Twitter (Zhuang et al., 2020), a computational bottleneck of applying a pipeline of NLP tools to a large text collection, and the limitations of lexical pattern matching.

In this work, we propose a simpler but more effective approach for automatically acquiring affective events by prompting pre-trained language models. We use one language model prompt to elicit affective event candidates, and we introduce a *Co-prompting* method to automatically label these event candidates with affective polarity. The key idea behind *Co-prompting* is to design two complementary prompts that capture independent views of an event, reminiscent of co-training (Blum and Mitchell, 1998). Combining information from two different views of an event produces labels that are more accurate than the labels assigned by either one alone.

Specifically, we acquire affective events in a two-step process: (1) Event Generation and (2) Polarity Labeling. The first step generates events that are associated with a set of gold "seed" affective events. For each seed event, we prompt a language model to generate sentences where the seed event co-occurs with some new events. Our hypothesis is that affective events are often preceded or followed by other affective events that are causally or temporally related. For example, if someone breaks his/her leg, a prior event might describe how it happened (e.g., *"fell off a ladder"* or *"hit by a car"*) and a subsequent event might describe the consequences (e.g., *"could not walk"* or *"rushed to the hospital"*).

The second step collects independent views of the polarity for each new event using two comple-

Findings of the Association for Computational Linguistics: ACL 2023, pages 3189–3201

mentary language model prompts. One prompt provides an *Associated Event View*, which considers the polarities of the known (labeled) events that co-occur with the new event during Event Generation. The second prompt provides an *Emotion View*, which considers the polarity of the most probable emotion words generated by a language model when prompted with the new event. Finally, we combine information from the two co-prompts to assign an affective polarity label to each new event.

Our experiments show that using these automatically acquired affective events as additional training data for an affective event classifier produces state-of-the-art performance over two benchmark datasets for this task. The analysis also confirms that our co-prompting method utilizing multiple views yields more accurate polarity labels than using either view alone.

In summary, the contributions of our work are:

1. We propose a method to generate weakly labeled data by prompting language models for the task of affective event classification. The method is effective but also simple, as it does not require fine-tuning any language model nor mining data from a text corpus.

2. We show that prompting for multiple views produces more accurate labels than prompting for a single view, as multiple views capture independent and complementary information.

2 Related Work

Several lines of research have recognized the importance of identifying events that carry affective polarity, including early work on plot units (Lehnert, 1981) and later work that learned patient polarity verbs (Goyal et al., 2010, 2013), emotion-provoking events (Vu et al., 2014), patterns associated with first-person affect (Reed et al., 2017), major life events (Li et al., 2014) and +/- effects for opinion analysis (Choi and Wiebe, 2014; Deng and Wiebe, 2014, 2015).

Recent work has focused specifically on classifying affective event phrases (Ding and Riloff, 2016, 2018; Saito et al., 2019; Zhuang et al., 2020). Ding and Riloff (2016) created a weakly supervised method for labeling events with affective polarity using label propagation. Ding and Riloff (2018) subsequently developed a method that assigns affective polarity to events by optimizing for semantic consistency over a graph structure, and created an Affective Event Knowledge Base (AEKB) of more than half a million event phrases labeled with affective polarity. Zhuang et al. (2020) later created an Aff-BERT classifier that substantially outperformed AEKB for affective event classification by training BERT with a relatively small amount of gold data. They additionally developed a Discourse-Enhanced Self-Training (DEST) method that further improved Aff-BERT's performance. Their approach used Twitter to collect events that corefer with sentiment expressions in specific lexical patterns. Our work also aims to improve a classification model with automatically generated affective events. The key difference is that our work produces weakly labeled affective events by prompting pre-trained language models, which alleviates the computational and practical problems of conventional pattern matching over a text corpus.

Pretrained language models such as GPT-2 (Radford et al., 2019) and BERT (Devlin et al., 2019) have been shown to learn diverse world knowledge (Petroni et al., 2019; Davison et al., 2019; Jiang et al., 2020; Talmor et al., 2020). Researchers have studied how to prompt language models to transfer their knowledge to downstream tasks, including prompt-based fine-tuning, automatic prompt search, and discrete/continuous prompt optimization (Shin et al., 2020; Qin and Eisner, 2021; Schick and Schütze, 2021a,b). Our work differs in several aspects. Essentially, our approach utilizes co-prompts to elicit multiple types of information (views) that are independent and complementary to each other. This is significantly different from prior work that used a single prompt or an ensemble of prompts that seek the *same* type of information.

Our work is also related to recent work on data augmentation using language models. For example, Anaby-Tavor et al. (2020) proposed a method, LAMBDA, that first fine-tunes GPT-2 over labeled data and then synthesizes weakly-labeled data. Kumar et al. (2020) proposed a similar but unified approach for pretrained transformer-based language models. (Yang et al., 2020) fine-tuned two different generative language models to generate questions and answers separately for reading comprehension. Most of these works require fine-tuning the language models while our approach does not.

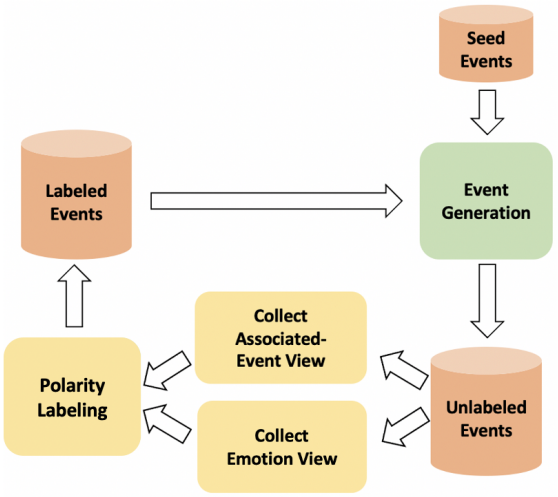

Figure 1: Flowchart for Acquiring Affective Events

3 Acquiring Affective Events with Multiple View Co-prompting

Our research aims to automatically generate labeled affective events to improve classifiers because gold data for affective event classification is only available in limited quantities. Automated methods for data generation offer a cost-effective and practical solution for improving the performance of affective event classifiers, and also could be used to rapidly acquire training data for new domains or text genres.

Figure 1 shows the flowchart for our approach. The process begins with a modest amount of "seed" data consisting of gold labeled affective events. The first step (**Event Generation**) uses a language model prompt to elicit events that are associated with each seed event. The second step (**Polarity Labeling**) assigns a polarity label to each new event using *Co-prompting* to assess polarity from two independent views of the event. Given an event e, the *Associated Event View* considers the affective polarities of labeled events that co-occur with e during Event Generation. The *Emotion View* considers the affective polarities of emotion words that are generated by an Emotion Prompt given the event e. Polarity scores produced from these views are then combined to assign an affective polarity label to the event e.

This process repeats in an iterative fashion, where the newly labeled events are used to discover more affective events in the next cycle. The process ends when no new events are generated or a maximum number of iterations is reached.

3.1 Event Generation

The Event Generation process begins with a set of gold affective events and produces a set of new events, many of which we expect to be affective. For each seed event, we create an **Associated Event Prompt** of the following form:

Here are the *{POLARITY}* things that happened to me today: *{EVENT}*,

where {EVENT} is a placeholder filled by the seed event phrase, and {POLARITY} is a placeholder filled by the affective polarity of the seed event. The prompt is designed to ask a generative language model to complete the sentence by enumerating other events that are likely to co-occur with the given event on the same day. The enumeration behavior is encouraged by the colon ':' and comma. The temporal relation is encouraged by the word 'today'. The polarity placeholder, {POLARITY}, encourages the language model to generate events with the same affective polarity.

For the polarity terms, we used the word *'good'* for events with positive polarity and the word *'bad'* for events with negative polarity. For events with neutral polarity, we simply used an empty string (i.e., *'Here are the things...'*).[1] We expected that this prompt would generate some neutral events, but that it would produce positive and negative events too because people tend to recount events that are interesting or impactful, not boring and mundane. In fact, we do not expect any of these prompts to be perfect. Our goal at this stage is to generate a healthy mix of new events across all three affective polarities (positive, negative, and neutral). The affective polarity for each new event will ultimately be determined later in the Polarity Labeling step.

To be consistent with prior work on this topic, we represent each event expression as a 4-tuple of the form: <Agent, Predicate, Theme, Preposition Phrase (PP)>. To create an event phrase for the language model prompt, we concatenate the words in the tuple. For example, given the negative event *<my house, burn down, -, ->*, the filled prompt would be *'Here are the bad things that happened to me today: my house burn down,'*.[2]

[1] We tried using some neutral words (e.g., 'neutral') in the placeholder, but we found that the empty string worked better.

[2] The event phrase may not be grammatically correct, but our observation is that this did not cause serious problems for the language model.

Polarity	Seed Event	Events Generated by Associated Event Prompt
NEG	I cut my leg	I fall off my bicycle, I hurt my knee, I wake up at hospital, I break my rib, I faint, kick me in head, they take my dog, my eye start to water, I break my ankle, I get in car accident
	I not get refund	they take my money, kick me out of game, this happen, freeze my account for hour, I lose money, I get refund, I get angry, ban me, make decision, I get email
	I lose my job	I break up with my girlfriend, I not apply, kick me out of house, arrest me, I go to find out, they try to kill me, I find job, eat my lunch, dump me, I break down
NEU	I walk in class	I start to talk to people, I take seat, I reply, professor tell me, my friend ask me, I take moment, I shake hand, I learn, I sit in front row, I have to explain, I want to tell story
	I close account	I call customer service, message say, I click on link, ban me for day, email tell me, I go, this show me, receive phone call, delete me, I call bank
	I meet someone	I get call from them, I get my drink, I lose weight, I chat for minute, I say something stupid, person tell me, I start to talk, I talk for long time, they invite me, they respect me
POS	I get in college	convince myself, I graduate, I go, I read them, drink coffee, I meet cool people, watch tv, I learn lot about myself, I move, I find good job
	I play match	my team win game, I lose, I go to hotel, I work, I go, I go on stage, I get score, I go to bed, play video game, I get point
	I get house	I pay my tax, I move out of my apartment, I eat my favorite food, I get new job, I start to live, I learn, I pay bill, I care, I afford to eat, I start to look

Table 1: Examples of events generated by the Associated Event Prompt for seed events

We used open-source GPT-2_{LARGE} (Radford et al., 2019) as the generative language model.[3]. To obtain diverse outputs, we let GPT-2 generate 200 sentences for each labeled event.[4] For the sampling method, we used nucleus sampling (Holtzman et al., 2020) with 0.9 as the top-p threshold, beam search with a beam size of 5, and a temperature of 2.0. We extracted new events from the sampled sentences to create event tuples, following the same conventions as earlier work (Ding and Riloff, 2018; Zhuang et al., 2020). For the sake of robustness, we selected the events that occur with at least 3 distinct seed events as new events for polarity labeling.

To illustrate, one example sentence generated from the event <*my house, burn down, -, -* > is " *..., my mom passed away and my family lost everything*.", and the events extracted are <*my mom, pass away, -, ->* and <*my family, lose, everything, ->* . We show more examples of extracted events in Table 1. Overall, the generated events are usually related to the seed event in some way and typically have the same affective polarity (e.g., "*I cut my leg*" → {*"I fall off my bicycle", "I hurt my knee", ...*}), despite some exceptions (e.g., "*they take my dog*"). For our purposes, it is perfectly fine that some generated events are loosely associated with the seed events, because our goal is simply to harvest new affective events, and their precise relationship to the seed events is irrelevant.

3.2 Polarity Labeling with Multiple Views

The next step is to assign affective polarity labels to each new event. We collect affective information from two prompts that provide independent views of an event: (1) we collect affective polarity information from the events generated by the *Associated Event Prompt*, and (2) we use the *Emotion Prompt* to generate emotion terms associated with an event. Finally, we combine the information gathered from these two prompts to assign a polarity label.

3.2.1 Emotion Prompting

To acquire another source of information about the affective polarity of an event, we prompt a language model to produce emotion terms with associated probabilities for each event. We design a cloze expression to generate emotion terms following an event expression by prompting a masked language model. Specifically, we use the following **Emotion Prompt**: **[*EVENT*]. I feel _ .**

The word "feel" leads the language model to return words that refer to emotions or other sentiments. We expect that positive events will typically be followed by positive emotions, and negative events by negative emotions. For neutral events, we expect to see a mix of both positive and negative emotions because these events can occur in a wide variety of contexts. We used BERT$_{\text{LARGE}}$ (Devlin et al., 2019) as the masked language model.[5] We store all generated terms and their probabilities produced by BERT for later use.

Figure 2 illustrates this process for two example

[3]Code available at https://github.com/openai/gpt-2

[4]We discarded samples that did not end with a period.

[5]We also experimented with using GPT-2 to generate emotions, but it was less effective and often produced sentences rather than emotion words, such as "*I break my arm. I feel like this is a real thing.*"

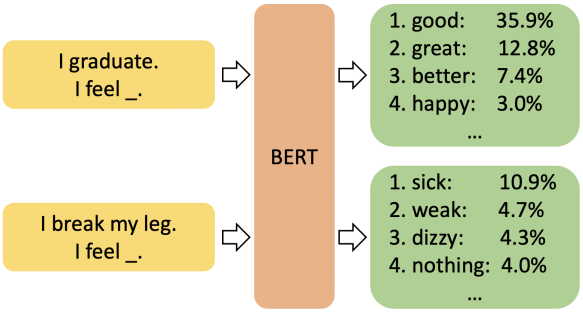

Figure 2: Emotion Prompt examples

events. The top shows the four most probable terms generated from the event tuple *<I, graduate, -, ->*, all of which have positive polarity. The bottom shows the four most probable terms generated from the event tuple *<I, break, my leg, ->*. Three of these terms have negative polarity, but the fourth term has neutral polarity. This example shows that the prompt can produce inconsistent results, but the probability distribution across all of the generated terms typically captures a fairly reliable signal.

3.2.2 Multiple View Polarity Scoring

We first define scoring functions to determine the most likely affective polarity for an event from each view independently. Then we present a joint scoring function that combines the scores from the two views to produce a final affective polarity label.

Associated Event View This view captures the degree to which an event co-occurs with labeled events of each polarity. Intuitively, we expect that events tend to co-occur with other events of the same polarity. According to this view, we define the *Associated Event Score* (S_A) of an unlabeled event e with respect to a polarity label l as:

$$S_A(l \mid e) = \frac{\sum_{e' \epsilon AEP(e)} I(e', l)}{|AEP(e)|} \quad (1)$$

where $AEP(e)$ is the set of labeled events that co-occur with e in the results produced by the Associated Event Prompt, $I(e', l)$ is an indicator function with a value of 1 if the polarity label of e' is l or zero otherwise, and $|\cdot|$ is the cardinality.

Emotion View This view captures the polarity of the emotion words generated by the Emotion Prompt. Based on this view, we define the *Emotion Score* (S_E) for an unlabeled event e with respect to a polarity label l as:

$$S_E(l \mid e) = \frac{\sum_{w \in D_l} P_{\text{BERT}}(w \mid EP(e))}{\sum_{l' \in L} \sum_{w \in D_{l'}} P_{\text{BERT}}(w \mid EP(e))} \quad (2)$$

where D is a gold dictionary of emotion terms, D_l is the subset of words in D that have polarity label l, and $P_{\text{BERT}}(w \mid EP(e))$ is the probability associated with word w produced by the Emotion Prompt (EP) given event e. For the gold dictionary D, we collect all of the adjectives and nouns in the MPQA subjectivity lexicon (Wilson et al., 2005) along with their polarity labels.

Polarity Assignment We conservatively assign positive and negative polarities to an event only when both S_A and S_E predict the same polarity. Formally, we label an event e with polarity l when both scores for l exceed a confidence threshold θ.[6]

- if $S_A(pos \mid e) \geq \theta$ and $S_E(pos \mid e) \geq \theta$, then e is positive
- if $S_A(neg \mid e) \geq \theta$ and $S_E(neg \mid e) \geq \theta$, then e is negative

For the neutral polarity, we found that the emotion scores $S_E(neu \mid e)$ are low in most cases because the Emotion Prompt tends to generate emotional words even for neutral events. However, we observed that the Emotion Prompt is more likely to generate a mixed set of both positive and negative emotion words for neutral events, presumably because neutral events can occur in both types of contexts. Therefore we assign neutral polarity by looking for a small difference between the positive and negative emotion scores. Specifically, we consider an event e to be **neutral** based on both its neutral Associated Event Score $S_A(neu \mid e)$ and the absolute difference between its positive and negative Emotion Scores, $S_E(pos \mid e)$ and $S_E(neg \mid e)$:

- if $S_A(neu \mid e) \geq \theta$ and $1 - |S_E(neg \mid e) - S_E(pos \mid e)| \geq \theta$, then e is neutral

As an example, consider an event with $S_E(neg \mid e) = .50$ and $S_E(pos \mid e) = .40$, then $1 - |S_E(neg \mid e) - S_E(pos \mid e)| = .90$, which indicates that the event is very likely to be neutral. In our experiments, we set all θ values to be .90 based on the performance over the development set.

[6]Note that θ must be greater than 0.5 to avoid multiple label assignments to an event.

4 Evaluation

4.1 Datasets

We conducted experiments over two previously used datasets for affective event classification: (1) the **BLOG** dataset constructed by Ding and Riloff (2018), which contains 1,490 manually annotated events (20% Positive, 18% Negative and 62% Neutral) extracted from blog posts, and (2) the **TWITTER** dataset developed by Zhuang et al. (2020), which contains 1,500 manually annotated events (29% Positive, 23% Negative and 48% Neutral) extracted from Twitter. We performed 10-fold cross-validation on each dataset (8 folds for training, 1 fold for development, and 1 fold for testing).

4.2 Generating Newly Labeled Events

To generate newly labeled events for each domain (TWITTER and BLOG), we used the training data as the seed events and ran the process for 15 and 10 iterations, respectively. We chose these stopping points because they produced around 10,000 new events for each domain, and we wanted to keep the number of new events manageable. Between iterations, we added the maximum number of newly labeled events that would maintain the original data distribution of affective polarities.

Figure 3 shows the number of new events acquired for each iteration. Both curves start at around 1,200 because that is the size of the gold training sets used for seeding. This process ultimately produced (on average, across the folds in our cross-validation experiments): 10,636 new events for the TWITTER domain and 10,800 new events for the BLOG domain.

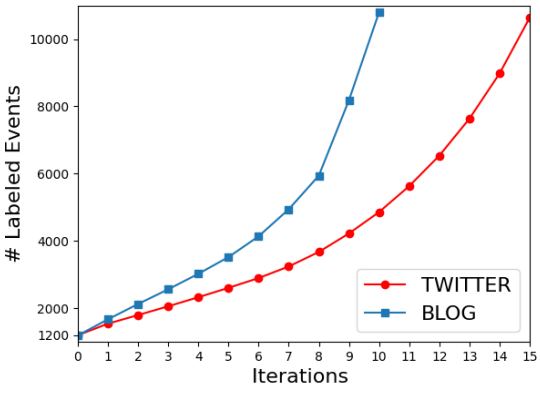

Figure 3: Newly labeled events generated across iterations.

4.3 Affective Event Classification Model

We use Aff-BERT (Zhuang et al., 2020) as our classification model, which is an uncased BERT-base model (Devlin et al., 2019) that takes an event tuple as input (we concatenate all of the words into a phrase) and classifies the phrase with respect to three affective polarities (positive, negative, or neutral). We train Aff-BERT with a weighted cross-entropy function, which weights the gold and the new (weakly) labeled data differently: $L = L_G + \lambda L_W$, where L_G is the loss over the gold data, L_W is the loss over the weakly labeled data, and λ is a weight factor. During training, we performed a grid search over all combinations of learning rates (1e-5, 2e-5, 3e-5), epochs (5, 8, 10), batch sizes (32, 64), and λ (0.1, 0.3, 0.5). We used the values that performed best over the development set.

4.4 Comparisons with Prior Work

We compared our method with several other approaches. Three methods were previously proposed by Zhuang et al. (2020) for affective event classification : **1)** the Aff-BERT model; **2)** Aff-BERT with self-training; **3)** Aff-BERT with Discourse-Enhanced Self-Training (DEST). The latter two methods improve Aff-BERT by providing additional weakly labeled data. Since the DEST method is specific to Twitter, we only evaluated that approach on the TWITTER dataset.

We also evaluated two general-purpose methods for data augmentation: **4)** Back-translation (Sennrich et al., 2016), which generates paraphrases of an input phrase via machine translation, and **5)** pattern-exploiting-training (PET) (Schick and Schütze, 2021a), which trains an ensemble of language models with multiple prompts and weakly-labeled data. For Back-translation, we translated each event phrase from English to German and then from German back to English using the wmt19-en-de and wmt19-de-en machine translation models released by Facebook (Ng et al., 2019). We then paired the output phrase with the original event's polarity label.

To train PET[7], we used BERT$_{BaseUncased}$ as the language model and used 3 prompts: *"[EVENT]. I feel _."*, *"[EVENT]. I felt _."* and *"[EVENT]. It was _."*. For hyperparameters, we used 1e-5 as the learning rate, 4 as the batch size, and 5 as the number of training epochs[8]. Since PET requires

[7] See https://github.com/timoschick/pet.

[8] We selected the hyperparameters using development data.

unlabeled data, we used 20K events randomly collected from the Affective Event Knowledge Base produced by Ding and Riloff (2018) for experiments with the BLOG data, and we used the 8,532 unlabeled events released by Zhuang et al. (2020) for experiments with the TWITTER data.[9]

4.5 Experimental Results

Tables 2 and 3 show our experimental results, including the precision (Pre) and recall (Rec) for each polarity as well as macro-averaged F1 scores. The *Aff-BERT* row shows the results when trained over only gold labeled data. The other models exploit weakly labeled data for additional training.

Method	Macro F1	POS		NEG		NEU	
		Pre	Rec	Pre	Rec	Pre	Rec
Aff-BERT	75.7	74.4	71.5	79.0	74.0	76.1	80.1
Back-translation	76.4	80.4	69.2	79.2	75.1	75.3	83.4
Self-training	77.0	78.6	69.5	76.8	**82.3**	77.4	79.8
PET	78.3	78.1	75.6	78.2	81.6	79.2	79.1
DEST	79.0	81.8	74.8	78.4	80.0	79.4	82.4
Co-prompting	**81.3**	**82.3**	**76.2**	**85.9**	79.7	**79.7**	**86.1**

Table 2: Experimental results for TWITTER data.

On the TWITTER data, Co-prompting outperforms all other methods. We see a 5.6% absolute F1 score gain compared to Aff-BERT and a 2.3% gain compared to DEST, which is the strongest competitor. Most notably, we see a 3.7% recall gain over DEST for neutral polarity and a 7.5% precision gain for negative polarity.

Method	Macro F1	POS		NEG		NEU	
		Pre	Rec	Pre	Rec	Pre	Rec
Aff-BERT	77.4	71.7	66.2	78.2	77.2	85.0	87.4
Back-translation	77.9	79.6	66.1	75.5	74.3	85.3	90.0
PET	78.0	78.5	60.2	81.4	**76.5**	83.8	91.1
Self-training	78.6	76.3	68.3	78.6	76.2	**85.5**	89.0
Co-prompting	**80.7**	**81.4**	**70.1**	**84.0**	75.3	85.4	**91.8**

Table 3: Experimental results for BLOG data.

On the BLOG data, Co-prompting also consistently outperforms the other methods. It surpasses Aff-BERT by 3.3 absolute points in F1 score, and self-training (the closest competitor) by 2.1 absolute points. In addition, it achieves the highest precision for both positive and negative polarity.

4.6 Impact of Multiple Views

We also conducted experiments on the TWITTER data to understand the contribution of each view for polarity labeling.

[9]The AEKB data could be found at https://github.com/yyzhuang1991/AEKB and the unlabeled data for TWITTER could be found at https://github.com/yyzhuang1991/DEST

Method	Pre	Rec	F1
Emotion View	78.9	78.3	78.2
Associated Event View	79.4	78.9	78.8
Both (Co-prompting)	82.6	80.7	81.3

Table 4: Impact of Multiple Views on TWITTER Data

Table 4 shows the performance of models trained with events labeled by each view alone and by both of them together. Each view performs well on its own and produces classification models that outperform Aff-BERT. But Co-prompting yields a substantially higher F1 score than either view on its own.

Next, we investigated how and why the polarity labels change when incorporating both views. Figure 4 shows the number of labels that are changed correctly or incorrectly when adding the second view. The left table shows labels produced by the Associated Event View (AEV) that are changed by Co-prompting. For example, there are 19 good changes (wrong before, correct now) from neutral to negative (Neu → Neg) but 8 bad changes (correct before, wrong now). The Δ column shows the overall net gain in correct labels. Overall, Co-prompting has the greatest impact by correctly changing neutral labels to be positive or negative. This makes sense because the Associated Event View sometimes had trouble recognizing affective polarity, but the Emotion View specifically tries to identify emotions for each event.

AEV → Co	✔	✘	Δ
Neu → Neg	19	8	11
Neu → Pos	24	17	7
Pos → Neu	33	28	5
Neg → Neu	18	13	5
Pos → Neg	3	2	1
Neg → Pos	2	5	-3

EV → Co	✔	✘	Δ
Neu → Neg	23	7	16
Neg → Neu	29	14	15
Pos → Neu	38	24	14
Neg → Pos	4	3	1
Pos → Neg	0	1	-1
Neu → Pos	22	23	-1

Figure 4: Counts of labels changed by Co-prompting (Co). ✔: correct. ✘: incorrect. Δ: correct - incorrect

The table on the right side of Figure 4 shows labels produced by the Emotion View (EV) that are changed by Co-prompting. Adding AEV has the greatest impact in the opposite direction: changing mislabeled negative or positive events to be neutral. Intuitively, this is because EV can be too aggressive about assigning positive and negative polarity and have difficulty recognizing neutral events. These results nicely illustrate the power of Co-prompting: complementary views have different strengths and

weaknesses, and the strengths of one view can compensate for weaknesses in the other. And more generally, Figure 4 shows that most of the label changes produced by Co-Prompting were more accurate than the labels produced by one view alone, demonstrating that Co-Prompting with complementary views adds robustness.

4.7 Manual Analysis

To directly assess the accuracy of the polarity labels assigned by Co-prompting for the newly generated events, we asked two people to annotate 200 randomly sampled events from TWITTER.[10] The pairwise inter-annotator agreement was 89.5% using Cohen's kappa. The annotators then adjudicated their disagreements.

Polarity	AEV	EV	Both
POS	50/62 (80.6%)	50/58 (86.2%)	62/68 (**91.2%**)
NEG	33/40 (82.5%)	43/49 (87.8%)	33/35 (**94.3%**)
NEU	85/98 (86.7%)	73/93 (78.5%)	87/97 (**89.7%**)
Overall	168/200 (84.0%)	166/200 (83.0%)	182/200 (**91.0%**)

Table 5: Manual Analysis of Polarity Labels

Table 5 shows the accuracy of the labels produced by each view alone and by Co-prompting (Both). The overall accuracy is only 83%-84% for the labels produced by each view but 91% for the labels produced by both views. The Associated Event View is most accurate for neutral labels, whereas the Emotion View is most accurate for positive and negative labels. These results again confirm the value of complementary sources of information for labeling data.

4.8 Learning Curves

We produced learning curves to understand the behavior of training with different amounts of data on the TWITTER domain. Figure 5 plots the F1 scores of Co-prompting when re-training the classification model with the data generated after every 3 iterations. The dashed line shows the F1 score of Aff-BERT (using only gold data) for comparison. The F1 score of Co-prompting rises steeply after the first 3 iterations, and continues to improve across later iterations. This graph suggests that running the iterative process even longer could yield further benefits.

We also investigated the effectiveness of our approach with smaller amounts of gold seed data.

[10]They followed the same annotation guidelines used to create the TWITTER and BLOG datasets as defined by (Ding and Riloff, 2018).

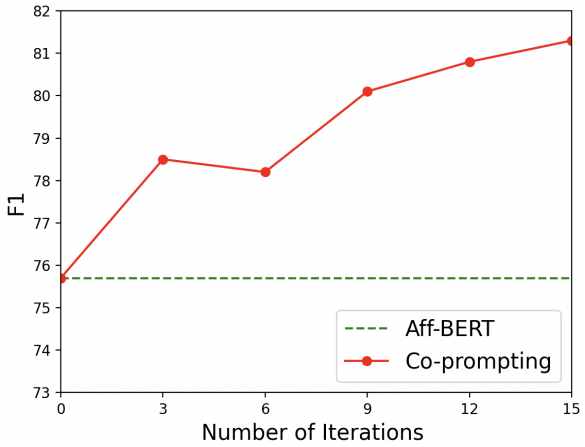

Figure 5: Learning curve of Co-prompting

Figure 6 shows the performance of Co-prompting on the TWITTER data when trained with subsets of the gold data ranging from 50% to 90%. For comparison, we also show the results for the two strongest competitors, DEST and PET, as well as the Aff-BERT baseline[11]. Co-prompting consistently outperforms the other approaches over all training set sizes. Surprisingly, Co-prompting trained with only 50% of the gold data achieves the same level of performance as Aff-BERT using 100% of the gold data. This result demonstrates that generating labeled events with our co-prompting method can produce a high-quality classification model even with smaller amounts of gold seed data.

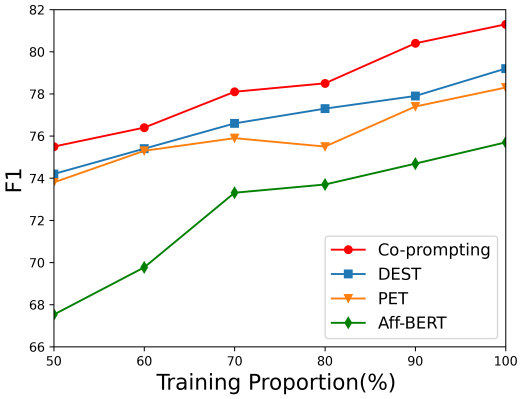

Figure 6: Results for different training set sizes

5 Conclusions

We presented a novel approach for eliciting and labeling affective events by co-prompting with

[11]The results of Aff-BERT and DEST are reported in (Zhuang et al., 2020)

large language models. Our approach does not require fine-tuning and is more practical than pattern-matching over large text collections, as has been done in prior work. The key idea is to design complementary prompts that collect independent types of information, which can then be used jointly as weak supervision to robustly label new data. Our experimental results show that labeling with multiple views is highly effective and that the elicited events substantially improve an affective event classifier.

Finally, we believe that co-prompting is a general idea that should be applicable for other data harvesting tasks as well. Co-prompting is more robust than relying on just one type of information, and we hope that other researchers will explore this idea for different types of NLP problems.

6 Limitations

We presented a method to automatically generate and label affective events by co-prompting with large language models. The data generation process does not involve creating or training new language models. There are some limitations to our approach. One limitation is that language models are not guaranteed to generate truthful or sensible information, which could introduce noisy information to our model. For example, we observed that the Emotion Prompt sometimes generates highly unlikely polarity labels for some events. Language models can also produce biased results, which could introduce biased information to our model. Another limitation is that it may be non-trivial for researchers who want to apply our method to other NLP problems to design prompts that are effective for their task. We believe that this method should be fairly general, but it has not yet been evaluated for other tasks. Lastly, our method requires a moderate amount of computational resources, including GPU cards with substantial memory and access to large language models. As a result, groups with limited resources might find our method too computationally intensive.

Acknowledgement

We thank Tianyu Jiang for his helpful comments on our work. We also thank the anonymous reviewers for their insightful feedback.

References

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, N. Tepper, and Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! In *Proceedings of the Twentieth AAAI Conference on Artificial Intelligence (AAAI 2020)*.

A. Blum and T. Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98)*.

Yoonjung Choi and Janyce Wiebe. 2014. +/-EffectWordNet: Sense-level Lexicon Acquisition for Opinion Inference. In *Proceedings of the 2014 Conference on Empirical Methods on Natural Language Processing (EMNLP 2014)*.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1173–1178, Hong Kong, China. Association for Computational Linguistics.

Lingjia Deng and Janyce Wiebe. 2014. Sentiment Propagation via Implicature Constraints. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*.

Lingjia Deng and Janyce Wiebe. 2015. Joint Prediction for Entity/Event-Level Sentiment Analysis using Probabilistic Soft Logic Models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT/NAACL 2019)*.

Haibo Ding and Ellen Riloff. 2016. Acquiring Knowledge of Affective Events from Blogs using Label Propagation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016)*.

Haibo Ding and Ellen Riloff. 2018. Weakly Supervised Induction of Affective Events by Optimizing Semantic Consistency. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*.

A. Goyal, E. Riloff, and H. Daumé III. 2010. Automatically producing plot unit representations for narrative text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*.

Amit Goyal, Ellen Riloff, and Hal Daumé III. 2013. A Computational Model for Plot Units. *Computational Intelligence*, 29(3):466–488.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.

Wendy G Lehnert. 1981. Plot Units and Narrative Summarization. *Cognitive Science*, 5(4):293–331.

Jiwei Li, Alan Ritter, Claire Cardie, and Eduard Hovy. 2014. Major life event extraction from twitter based on congratulations/condolences speech acts. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP 2014)*.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 314–319, Florence, Italy. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Lena Reed, JiaQi Wu, Shereen Oraby, Pranav Anand, and Marilyn A. Walker. 2017. Learning lexico-functional patterns for first-person affect. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*.

Jun Saito, Yugo Murawaki, and Sadao Kurohashi. 2019. Minimally Supervised Learning of Affective Events Using Discourse Relations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP/IJCNLP 2019)*.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235. Association for Computational Linguistics.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2020. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Hoa Trong Vu, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. 2014. Acquiring a Dictionary of Emotion-Provoking Events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages

1008–1025. Association for Computational Linguistics.

Yuan Zhuang, Tianyu Jiang, and Ellen Riloff. 2020. Affective event classification with discourse-enhanced self-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5608–5617. Association for Computational Linguistics.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Sec 6.

☒ A2. Did you discuss any potential risks of your work?
There is no potential risk, to our best knowledge, in this research topic.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
the abstract section and Sec 1.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

we use some pretrained language models which are mentioned in Sec 3 and Sec 4. The dataset we used is mentioned in Sec 4.

☑ B1. Did you cite the creators of artifacts you used?
We cite the artifacts in Sec 3 and Sec 4.

☒ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
The artifacts I used are open to the research community, so we don't think we need to discuss it.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
The artifacts I used have no intended use specified. We used them for fine-tuning (BERT), inference (BERT, GPT2) and evaluation (the datasets) only.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The data do not contain anything that uniquely identifies people or offensive content.

☒ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
The artifacts I used are pretty well known in the community (BERT, GPT2). And I don't think there is any explicit things like linguistic phenomena or demographic groups related to these models. I do mention some characteristics of the dataset (e.g., domain) I used in Sec 4.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Sec 4.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

Sec 4.

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
The models we used are pretty well known (BERT, GPT2). So we don't think we need to report these.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Sec 4.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Sec 4.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Sec 3 and Sec 4.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Sec 4.

☒ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
We don't think it is necessary to provide the instruction, as the instruction given to the participants is simply to judge if the model prediction is correct using their best knowledge.

☒ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
The annotators are not recruited but our labmates in the lab.

☒ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
The data I collected is to evaluate how good our model is. There is no other use.

☒ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
There is no ethic concern in my task.

☒ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Our annotation task involves only commonsense judgement, in which basic demographic and geographic characteristics barely play any role.