Event Semantic Classification in Context

Anonymous ACL submission

Abstract

In this work, we focus on a fundamental yet underexplored problem, event semantic classification in context, to help machines gain a deeper understanding of events. We classify events from six perspectives: modality, affirmation, specificity, telicity, durativity, and kinesis. These properties provide essential cues regarding the occurrence and grounding of events, changes of status that events can bring about, and the connection between events and time. To this end, this paper introduces a novel bilingual dataset collected for the semantic classification tasks and models designed to address them as well. By incorporating these event properties into downstream tasks, we demonstrate that understanding the fine-grained event semantics benefits event understanding and reasoning via experiments on event extraction, temporal relation extraction and subevent relation extraction.

Context: The community warmly RECEIVED the refugees.
Event: RECEIVED
Synset of event: receive.v.5
Definition of synset (gloss): express willingness to have in one's home or environs.

Properties of RECEIVED
Modality: *realis*
Affirmation: *affirmative*
Specificity: *specific*
Telicity: *telic*
Durativity: *durative*
Kinesis: *non-static*

Figure 1: An example of event semantic classification from six perspectives. The synset of the event is drawn from WordNet (Miller, 1992).

1 Introduction

A semantic class contains words that share a semantic feature. For example, within nouns, there are two subclasses, concrete nouns, and abstract nouns. Concrete nouns include people, plants, and animals, while abstract nouns refer to concepts such as qualities, actions, and processes. In this work, instead of classifying nouns that are rather comprehensible lexemes in text, our focus is on the **semantic classification of *events***. We perform semantic classification from multiple perspectives, which yields properties that are beneficial to comprehensive event understanding and relevant downstream tasks such as event extraction (Doddington et al., 2004; Wang et al., 2020b), event-event relation extraction (Glavaš et al., 2014; O'Gorman et al., 2016), and event reasoning (Han et al., 2021).

Different from conventional span classification tasks such as entity typing (Mikheev et al., 1998; Yaghoobzadeh and Schütze, 2015; Choi et al., 2018) and event typing (Walker et al., 2006; Wadden et al., 2019; Zhang et al., 2021) that map textual spans to predefined ontologies for abstraction purposes, we focus on understanding the fine-grained semantic qualities of an event. To facilitate this, we propose to classify events by their multi-faceted properties — modality, affirmation, specificity, telicity, durativity, and kinesis. The definitions of these properties are as follows[1]:

- Modality (actuality): whether an event actually occurs.
- Affirmation: whether an event is described affirmatively.
- Specificity (genericity): whether an event refers to a particular instance.
- Telicity (lexical aspect): whether an event has a specific endpoint.
- Durativity (punctuality): whether an event happens momentarily.

[1] Details about these properties are discussed in §2.

- Kinesis: whether an event describes a state or an action.

Among these properties, modality, affirmation, and specificity are of great help to understanding the occurrence and grounding of an event, since modality and affirmation indicate if an event actually occurs (Hopper and Thompson, 1980), whereas specificity indicates whether an event is understood as a singular occurrence, a finite set of such occurrences, or others (Doddington et al., 2004). Telicity and durativity, on the other hand, are properties that connect events with time, and thus they evidently provide useful cues for temporal reasoning in narrative text. And the last property, kinesis, divides events into states and non-states. Examples that belong to states include "desire," "want," "love," and so forth. They involve no dynamics and do not constitute changes themselves (Mourelatos, 1978).

There are a few works that have incidentally tagged some properties for events in the TimeML (Pustejovsky et al., 2003), ACE (Doddington et al., 2004), MASC (Ide et al., 2008), and UDS (Gantt et al., 2022) annotations. Yet only modality has been addressed with machine learning approaches in Monahan et al. (2015). In terms of usage of these properties, previous effort has been limited to leveraging them in feature-based statistical learning methods for the event coreference resolution task (Ahn, 2006; Bejan and Harabagiu, 2010). In a nutshell, we lack the tools to obtain these useful attributes and have not fully exploited them for event understanding and reasoning tasks.

In this paper, we introduce ESC, the first comprehensive dataset collected for event semantic classification in both English and Chinese. It contains all the WordNet (Miller, 1992) example sentences for frequent verbs that feature 5,015 eventive synsets. The event mentions within these sentences are annotated with their six semantic properties. We also introduce and evaluate several models for the proposed tasks. By incorporating the event properties predicted by our best model into multiple event-related tasks, we demonstrate the utility of these properties through detailed experimental analysis. The contribution of this paper is threefold:

- We introduce a new bilingual dataset for fine-grained event semantic classification tasks in English and Chinese.
- We design novel models for classifying events by six properties and evaluate the performance of large language models (LLMs) on this task.
- To enhance the model performance of event understanding, we propose a constraint learning and enforcing methodology for incorporating event properties and evaluate on three downstream datasets.

2 Event Properties

This section introduces six event properties we aim to address and why we choose them in detail. We also provide examples and analysis on how they assist event reasoning tasks.

2.1 Modality

Modality, also referred to as actuality, classifies events into realis and irrealis. Realis indicates that an event is *a statement of fact*, in other words, the event actually happens. For example, the "speak" event in "I hired an assistant who SPEAKS English" actually occurs. On the contrary, if the context of an event is expressing nonactual or nonfactual, then the modality of the event is irrealis. For example, the "speak" event in "I am looking for an assistant who SPEAKS English" is in an irrealis mode. The modality property of events presents the grounding and occurrence information. This is useful in event coreference resolution and temporal relation extraction since it is unreasonable to predict the coreferential or temporal relation between a non-factual event and an event that actually occurs.

2.2 Affirmation

Affirmation is similar to modality in the sense that they are both properties about the happening of an event. Affirmation divides events into those mentioned in affirmative clauses like "we e_1:HAD some bread yesterday" and those mentioned in negative clauses like "but now we e_2:HAVE no more bread." Yet different from modality, we can explore the temporal order between affirmative events and negative events, e.g., the temporal relation between (e_1, e_2) is BEFORE. Essentially, we use *realis* for statements of fact, either affirmative or negative, and *irrealis* for anything contrary to fact, either affirmative or negative. And this is why we separately handle affirmation and modality, instead of merging them into one event property, i.e., polarity in the ACE annotations (Doddington et al., 2004).

2.3 Specificity

There are specific events and generic events if we classify them with **specificity**. Generic events can be found in the following example: "After HAVING a large meal, lions may SLEEP longer." In contrast, the events in the following sentence, "the lion HAD a large meal and SLEPT for 24 hours," are both specific ones. We cannot infer any event relations across the two example sentences, given that events within different sentences do not agree on specificity with each other.

2.4 Telicity

Telicity describes how an event is structured in relation to time. If an event has a natural endpoint, it is said to be telic; if the situation an event describes is not heading for any particular endpoint, it is said to be atelic. A common example of events that differ in their lexical aspect is "arrive" and "run": the former has a natural endpoint while the latter does not. However, "run" in a certain context, like "RUNNING ten miles", has a natural endpoint. Another example is "I ATE it up" and "I am EATING it": the former activity is viewed as completed and telic, while the latter is atelic. Though we may determine the telicity for part of event triggers without any context, we can observe changes in telicity for event triggers in different contexts. And that is why we need to provide contexts of events when annotating telicity.

Some readers may argue that this "endpoint" testing for events is not clear enough, since any event, if placed in a longer time scale, would always have an endpoint. On that account, we consider another algebraic definition of telicity proposed by Krifka (1989): telic events are quantized, while atelic ones are cumulative. This would be easy to understand if we took a dimensionality increase perspective. We can view entities as objects in the three-dimensional space and events as objects in the four-dimensional space where time is introduced as an extra axis. Of course, events are different from entities in many ways, e.g., events often involve the interaction among multiple entities, yet a remarkable difference between entities and events is that events interact with time. Note that there is a countability distinction in the entity domain: "book," "chair," and "person" are countable, whereas "water," "food," and "air" are uncountable. If we apply the countability concept to the time axis in the event domain, we can get countable events (or telic events) like "SOLVE a puzzle" and uncountable events (or atelic events) like "WALK around aimlessly." With the help of the algebraic definition, the inter-annotator agreement (IAA) is significantly improved compared to when only the "endpoint" definition is given (see Tab. 1).

Telicity is beneficial to temporal reasoning in that it provides endpoint information about events. For instance, consider the following two sentences: "he e_3:RAN his eyes over her body and e_4:KISSED her on the forehead" and "he was in e_5:LOVE with her and e_6:KISSED her on the forehead." Notice that e_3:RAN in the first sentence is a telic event that has an endpoint whereas e_5:LOVE in the second is an atelic event that has no endpoint. Therefore, the temporal relationship between the first event pair (e_3, e_4) is BEFORE, and the temporal relation between the second pair (e_5, e_6) is INCLUDES.

2.5 Durativity

Durativity classifies events into two categories: durative events and punctual events. Punctual events are those that happen within several seconds, such as "KICK a football" and "LOSE my wallet"; and durative events last for some period of time longer than seconds: for instance, "GO to school" typically takes tens of minutes, and "LOSE weight" usually takes several months. Note that "lose" can be punctual and durative events in different contexts. So is the case for many other event triggers, and thus we need to study the durativity of events with contexts.

As shown in Zhou et al. (2020), the duration of events not only provides important cues in temporal reasoning but in event coreference and parent-child relations as well. It is evident that two events with different durativity features are not coreferential to each other. And a punctual event cannot be the parent of a durative event, given that a parent-child relation entails spatio-temporal containment.

2.6 Kinesis

Kinesis is a property that distinguishes states from non-states (actions). Non-static events usually bring about status changes in event participants, whereas static events do not. Continuing with the previous example "he was in e_5:LOVE with her and e_6:KISSED her on the forehead," e_5 is a state whereas e_6 is an action (non-state). Note that the kinesis of some event triggers can also be context-dependent, e.g., "own" is a non-state in the first example and a state in the second: (1) "he owned his mistake in front of the class," (2) "he owns

	Modality	Affirmation	Specificity	Telicity	Durativity	Kinesis
IAA	0.65	0.85	0.87	0.53	0.61	0.67

Table 1: Inter-annotator agreement (Fleiss' kappa) of the ESC annotation.

two houses." Based on the aforementioned three attributes, i.e., telicity, durativity, and kinesis, Comrie (1976) proposed to divide events into five categories as shown in Tab. 2. Here we do not dive deeper into the naming of event classes, since our focus is how they benefit event understanding and reasoning in general.

	Punctual	Durative
Telic	Achievement	Accomplishment
Atelic	Semelfactive	Activity
Static		State

Table 2: Comrie (1976)'s classification of events based on three properties: telicity, durativity, and kinesis.

3 Data Annotation

Though there are verbal and nominal events, we believe the learning of event properties for one class can be generalized to the other with the help of current LLMs. We select 2,416 verbs from the 5,000 most frequent words[2] in the Corpus of Contemporary American English (COCA). Regarding these verbs, there are 5,015 synsets and 7,399 example sentences in WordNet (Miller, 1992). We treat the example sentences as contexts of these verbal events. We translate the English context sentences into Chinese and extract the spans of verbs using their synsets' Chinese names in WordNet.

We employ the Data Collection and Labeling Services from Tencent Cloud[3] for our event property annotation, in which each assignment asks six questions regarding an event and costs ¥2.0 (∼$0.3). Each assignment takes about one minute to complete and the hourly payment is about $18. We require that our annotators are "Master Workers," indicating reliable annotation records. We identified 15 valid annotators: all of them are native Chinese speakers who have received higher education and speak fluent English. Before working on the annotation assignments, they are trained by experts to fully understand the instructions that provide definitions and examples of each event property (see §2)[4]. Each annotator is assigned 1,500 events such that each event is annotated by at least three annotators. The final labels are determined by majority voting and the IAA's (Fleiss' kappa) of the six tasks are shown in Tab. 1. We also provide sample annotation results in Tab. 3.

4 Classification Models

In this section, we introduce the models designed for the proposed classification tasks.

4.1 Multi-label Predictor

Given the context of an event, we first use a pretrained language model, XLM-RoBERTa (Conneau et al., 2020), to produce the contextualized embeddings for all tokens. To obtain the representation of the event h_e, we concatenate the hidden state of the last layer that is stacked on top of the event trigger e and the attention vector of the event. If the event trigger spans multiple subword pieces, the average of the subword representations is taken. We then use a multi-layer perceptron with six output logits followed by a sigmoid function to estimate the value for each property.

4.2 Indirect Supervision from Glosses

A gloss[5] provides the sense definition for a lexeme. For example, the gloss of "ran" in "He RAN his eyes over her body" is *pass over, across, or through*. With the gloss, the telicity of "ran" can be easily inferred as telic, since "pass over" has a natural endpoint. And here is another example in which gloss knowledge helps us determine the durativity of an event: the gloss of "touch" in "He could not TOUCH the meaning of the poem" is "comprehend." If we look at the trigger "touch" itself, we might think that it is somewhat punctual. However, the comprehension of a poem requires some careful reading and is actually a durative process that cannot be completed within seconds.

Given that gloss knowledge provides richer semantic information than the event trigger itself, we would like to leverage the glosses provided

[2]https://www.wordfrequency.info

[3]https://cloud.tencent.com/solution/data-collect-and-label-service

[4]The detailed guideline, annotation interface, and dataset statistics are shown in Appendix §8.

[5]We obtain the gloss of an event by looking up the definition of the synset of that event in WordNet.

Event in context	Modality	Affirmation	Specificity	Telicity	Durativity	Kinesis
He RAN his eyes over her body.	1	1	1	1	1	1
The setting sun THREW long shadows.	1	1	1	0	0	0
The community warmly RECEIVED the refugees.	1	1	1	1	0	1
Please PLUG in the toaster!	0	1	1	1	1	1
He could not TOUCH the meaning of the poem.	1	0	1	1	0	0
Lions only EAT meat.	1	1	0	1	0	1
He DEBUTS next month at the Metropolitan Opera.	0	1	1	1	0	1

Table 3: Sampled events (marked in BLUE) in context along with their annotated semantic properties. 1's and 0's respectively denote (Realis, Irrealis) for Modality, (Affirmative, Negative) for Affirmation, (Specific, Generic) for Specificity, (Telic, Atelic) for Telicity, (Punctual, Durative) for Durativity, (Action, State) for Kinesis.

by WordNet to enhance the model performances. Keeping the other components the same as our first model, we simply append the gloss to the beginning of the input context, e.g., "[CLS] Touch means comprehend in the following sentence. [SEP] He could not touch the meaning of the poem."

4.3 Few-Shot Learning with GPT-3

To evaluate the event understanding ability of GPT-3 (Brown et al., 2020), we design prompts and study event semantic classification in a few-shot fashion. As shown in Fig. 2, for each event property, we provide its definition and a few examples in the prompt, and ask GPT-3 binary questions about events. To overcome the commonly observed high variance issue of prompt-based approaches (Zhao et al., 2021), we set the number of examples even for each label (two examples each) to mitigate the majority label bias. We also conduct two sets of experiments by alternating the label of the last example[6], so as to mitigate the recency bias (outputting answers may be biased towards the end of the prompt). To make a fair comparison with the method proposed in §4.2, we also conduct another set of experiments by incorporating gloss knowledge into the prompt for each event.

Prompt: Telicity describes how an event is structured in relation to time. If an event has a natural endpoint, it is said to be telic; if the situation an event describes is not heading for any particular endpoint, it is said to be atelic. Below are a few examples.

Event: ran
Context: He ran his eyes over her body.
Telicity: telic

Event: threw
Context: The setting sun threw long shadows.
Telicity: atelic

Event: expecting
Context: We were expecting a visit from our relatives.
Telicity: atelic

Event: debuts
Context: This young soprano debuts next month at the Metropolitan Opera.
Telicity: telic

Please determine the telicity of the following event:
Event: flies
Context: Time flies like an arrow.
Telicity:
Response: atelic

Figure 2: An example prompt for GPT-3 to determine the telicity of an event in English. The text in apricot denotes the essential part of the prompt, whereas the other part contains definitions and examples of telicity which are excessive overhead information that could be reduced in the requests to ChatGPT.

4.4 Conversational Solution with ChatGPT

Recently, ChatGPT, which was trained with reinforcement learning techniques from human feedback, has drawn a huge amount of attention since it is able to interact with human beings and answer questions in broad domains. To see how well ChatGPT can perform on our tasks, instead of describing the event properties and examples in the prompt every time as what we do for GPT-3 (see Fig. 2), we exploit the advantage of the dialogue format of ChatGPT to reduce the excessive overhead. Specifically, we provide those additional information only at the first round of the conversation and ask binary questions regarding the event properties as follow-up questions. To mitigate the biases mentioned in §4.3, as well as to incorporate gloss knowledge, we conduct additional sets of experiments as counterparts of GPT-3 experiments.

5 Evaluation

In this section, we describe the experiments on the ESC dataset. We randomly 80/10/10 split the data into train/dev/test sets and use F_1 score as

[6]Basically we switch the last two examples in Fig. 2.

	Modality	Affirmation	Specificity	Telicity	Durativity	Kinesis	Avg.
MP	**0.95**	0.94	**0.95**	0.81	0.91	0.75	0.89
MP + Gloss	0.94	**0.96**	**0.95**	**0.84**	**0.93**	**0.80**	**0.90**
GPT-3	0.58	0.78	0.87	0.38	0.61	0.34	0.59
GPT-3 + Gloss	0.61	0.76	0.87	0.44	0.62	0.36	0.61
ChatGPT	0.65	0.73	0.92	0.40	0.66	0.35	0.62
ChatGPT + Gloss	0.66	0.79	0.89	0.51	0.69	0.42	0.66

Table 4: Experimental results on the ESC dataset (the numbers are averaged F_1 scores on English and Chinese). MP denotes the multi-label predictor, and MP+Gloss denotes the gloss-appended version of multi-label predictor. Bold number in each column denote the best result for each property.

the evaluation metric. For the multi-label predictor and its gloss-appended version, we select five random seeds to train the model and calculate the averaged F_1 scores on the test set. GPT-3 and ChatGPT-related results are averaged numbers of two different prompt settings on the test set.

We report the averaged F_1 scores on the English and Chinese test sets in Tab. 4. From the results we can see that the multi-label predictor with gloss knowledge offers the best performances in terms of F_1, outperforming the baseline multi-label predictor by 1% on average. It is notable that there is a 5% gain in the kinesis classification performance, given that MP+Gloss leverages both direct supervision from the labels and indirect supervision from gloss knowledge. GPT-3 and ChatGPT, with no direct supervision from the dataset, achieve decent performances of an average score of 0.59 and 0.62. With the help of gloss, we observe a 2% and 4% gain in the average performance across six event properties respectively for GPT-3 and ChatGPT.

Through the experiments, we find that the biggest problem of these large language models (LLMs) lies in that minor changes in the prompt can make huge differences in the response. For example, when we ask ChatGPT to determine the kinesis of "lay out" in the following sentence: "the nurse lays out the tools for the surgery," it gives different answers when the prompt varies from "Please determine the kinesis of the following event" to "Please determine the kinesis of the following event **and explain why**." With the first prompt, it is able to give the correct answer *non-static* ("lay out" in this context means to spread the tools out so that they can be easily accessible, which is obviously an action). However, when asked to provide an explanation, it first gives the opposite answer, *static*, and then provides the following explanation: "This is because the event is likely describing the act of arranging or organizing the tools, rather than involving any movement or change in the state of the tools or event participants." The first part of the explanation is correct, but from the second part, it seems that ChatGPT is not completely clear about the meaning of "change in state." Hence, how to improve the robust reasoning ability of LLMs requires further investigation.

6 Enhancing Event-Centric NLP Tasks

In this section, we leverage the event properties to improve the model performances on event reasoning tasks. We study two methods to this end, one is to incorporate these properties in existing models as features, and the other is to induce constraints and incorporate the constraints into the models. We examine three event-centric NLP tasks, namely event extraction, event temporal relation extraction, and subevent relation extraction, which serve as the media for demonstrating the effectiveness of our proposed tasks and models.

6.1 Event Extraction

Event extraction includes two subtasks, event trigger identification, and classification. Here we only focus on the classification part since we need to know the textual span of events first to determine their properties. Recent models for event extraction (Wadden et al., 2019; Lin et al., 2020) are mostly based on the tokens' contextual representations learned by pretrained language models. The event representations are then fed into neural networks to predict the event types in some predefined ontology. By concatenating the six-dimensional vector of event properties with event representations, we can easily add the semantic classification results as features. As another way of incorporating event properties, we leverage the semantic meaning of event types to induce constraints. For example, if an event has type TRANSPORT (a subtype of MOVEMENT) in ACE annotations (Doddington et al., 2004), then its durativity can only be *durative*. Similarly, if an event is subsumed under the

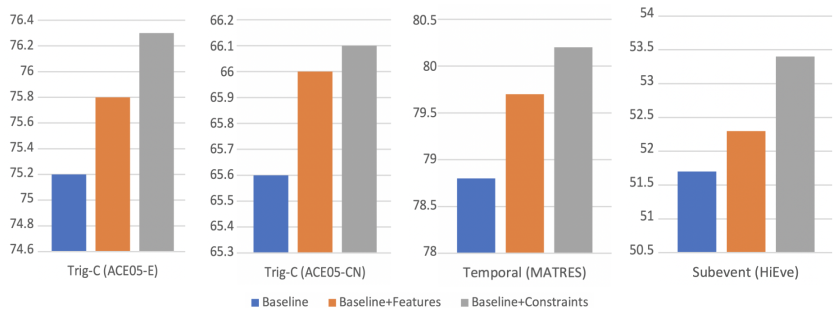

Figure 3: Experimental results of incorporating event properties in existing models. Trig-C is short for event trigger classification. Note that the baseline model for Trig-C is OneIE (Lin et al., 2020) while the baseline for the rest two is JCL (Wang et al., 2020a). The metric we use for all evaluations is F_1 score.

type of MEET (a subtype of CONTACT), then its kinesis can only be *non-static*.

Inspired by the expressiveness of Rectifier Network (Pan and Srikumar, 2016), we employ it to automatically learn constraints using the training set of ACE. Specifically, the constraints serve as criteria for whether an event with certain properties can belong to certain types. Let $\mathbf{X}_p$ be the property vector with six dimensions and $\mathbf{X}_t$ be the one-hot type vector (following Wadden et al. (2019)'s pre-processing method for ACE05-E and ACE05-CN dataset). Then the information to be included in the constraints about an event can be expressed as:

$$\mathbf{X} = \mathbf{X}_p \cup \mathbf{X}_t. \tag{1}$$

Let $\mathbf{Y}$ denote whether an event with properties $\mathbf{X}_p$ can be classified as event type $\mathbf{X}_t$. We obtain all the events with their types from the training set documents, and leverage our MP+Gloss model to predict the value of $\mathbf{X}_p$ for each event. We set the labels for these events to $\mathbf{Y} = 1$ (which are treated as positive examples). After we acquire all the possible $\mathbf{X}$ values, we randomly perturb the bits of positive examples to generate the same amount of negative examples and set the labels for those instances as $\mathbf{Y} = 0$. We represent the constraints for event-type classification as K linear inequalities where we assume K is the upper bound for all the rules to be learned. And $\mathbf{Y} = 1$ if $\mathbf{X}$ satisfies constraints c_k for all $k = 1, \cdots, K$. The k^{th} constraint c_k is expressed by a linear inequality:

$$\mathbf{w}_k \cdot \mathbf{X} + b_k \geq 0, \tag{2}$$

whose weights $\mathbf{w}_k$ and bias b_k are learned. Since a system of linear inequalities is equivalent to a Rectifier Network (Pan et al., 2020), we adopt a two-layer Rectifier Network for learning constraints

$$p = \sigma\Big(1 - \sum_{k=1}^{K} (\mathbf{w}_k \cdot \mathbf{X} + b_k)\Big), \tag{3}$$

where p denotes the possibility of $\mathbf{Y} = 1$ and $\sigma(\cdot)$ denotes the sigmoid function. We train the parameters $\mathbf{w}_k$'s and b_k's of the Rectifier Network in a supervised fashion. After obtaining the parameters, we fix them and add the constraints as a regularization term in the loss function (i.e., cross-entropy loss) of the OneIE model (Lin et al., 2020). Specifically, p is converted into the negative log space which is in the same space as the cross-entropy loss (Li et al., 2019). In this way, the loss corresponding to the learned constraints is

$$L_{cons} = -log\Big(\sigma(1 - \sum_{k=1}^{K} ReLU(\mathbf{w}_k \cdot \mathbf{X} + b_k))\Big). \tag{4}$$

6.2 Event-Event Relation Extraction

Event-event relation extraction is another set of tasks that require reasoning over event semantics. We study two tasks, namely event temporal relation extraction and subevent relation extraction in this work. Similar to how we add event properties into the event type classification model, we adopt two approaches here as well. One is to concatenate the event properties with event representations, and the other is to induce and integrate constraints into the learning objectives of the model. We follow the same process to obtain the positive and negative examples for constraint learning introduced in (Wang et al., 2021). We employ the joint constrained learning (JCL) model proposed by Wang et al. (2020a) to address the two tasks at the same time. Given that the training objective of JCL is a combination of annotation loss, symmetry loss, and transitivity loss, we directly add the constraints learned with Rectifier Network (see Eq. 3) into the loss function.

6.3 Experiments and Analysis

For event trigger classification, we follow the same training methodology proposed in (Lin et al., 2020) and evalutate on ACE05-E and ACE05-CN. While for event-event relation extraction, we adopt the joint training approach introduced in (Wang et al., 2020a) and evaluate on the MATRES and HiEve dataset. F_1 scores are used for evaluating the models' performances and the results are shown in Fig. 3. Adding event properties as feature vectors brings about significant improvement in the task of subevent relation extraction, outperforming the baseline model by relatively 2.5%. They also enhance the model performance via constraints learned by Rectifier Network. This is most notable in the task of event trigger classification, where the model performance is improved by relatively 1.9%. Overall, incorporating event properties via constraints works better than adding them directly to the event representations. This demonstrates that inducing and enforcing constraints in such ways better captures the inter-dependencies between different event properties, as well as their connection with event types and relations. And this also provides an effective paradigm to integrate useful semantic information into recent neural models.

7 Related Work

The study of event semantics has been the focus of both linguistics and philosophy for a long time. Early effort on this topic dates back to sixty years ago: Vendler (1957) classified verbal events into four categories on whether they express "activity," "accomplishment," "achievement" or "state." And the criteria for distinguishing "accomplishment" and "achievement" from the other two is they have certain endpoints, i.e., they are telic. Later, Comrie (1976) introduced durativity and kinesis to further categorize events into five classes (see Tab. 2). Though there are further efforts that classify events in finer ways (Bach, 1986; Moens and Steedman, 1988), this paper focuses on how semantic classification of events supports the understanding of event-centric reasoning tasks. The most relevant work to our focus are the ten different event facets involved in the transitivity property of a clause (Hopper and Thompson, 1980) and the seven attributes designed for examining eventiveness (Monahan and Brunson, 2014) (i.e., to determine whether a lexeme can be identified as an event). Annotated on the MASC corpus (Ide et al., 2008), the SitEnt dataset (Friedrich and Palmer, 2014; Friedrich et al., 2016) captures event vs. state distinctions. The DIASPORA dataset (Kober et al., 2020) annotates phone conversations for stativity and telicity. Nevertheless, these previous works have mainly established theoretical frameworks for event study and left building tools for machine reasoning as the future endeavor.

Recent efforts in event annotations have been made in event detection (Walker et al., 2006; Wang et al., 2020b), and event-event coreferential, temporal, hierarchical, and causal relations (Bejan and Harabagiu, 2010; Pustejovsky et al., 2003; Glavaš and Šnajder, 2014; Mirza and Tonelli, 2014). These corpora have enabled data-driven models to gain understanding of event semantics and how they interact with other events. However, models learned from these corpora often rely on dataset statistics (Wang et al., 2022b,a) and thus are biased towards prior knowledge and have limited interpretability.

8 Conclusion

In this work, we first study six event properties that help machines gain a deep understanding of events and then introduce a novel dataset we collect for event semantic classification[7]. Various semantic information can be inferred from these properties in that they provide the occurrence and grounding of events and their connection with time as well. We design six methods for event semantic classification, four of which involve recent large language models. Experimental results demonstrate that ChatGPT performs better than GPT-3 even though its response is still subject to minor perturbation of the prompt formats. On average, the model MP+Gloss performs best in the proposed tasks and it is employed to predict event properties in three downstream tasks. To enhance the performances of neural models proposed for these tasks, we discuss two methodologies for incorporating useful event properties. Results show that the predicted event properties are effective in enhancing the performances of existing models across three different tasks. Therefore, we claim that the fundamental task of event semantic classification benefits both event understanding and reasoning.

[7] We will release the data and code upon acceptance.

Limitations

This work builds on human annotations and the application of state-of-the-art language models. The models might be biased towards the corpus used for training. And we only use XLM-RoBERTa to acquire the representations of events in MP and MP+Gloss; there might be more powerful architectures. The training of our models requires GPU resources which might produce environmental impacts, though the inference stage does not take up much computational resources.

Ethics Statement

There are no direct societal implications of this work, though the dataset we introduce in this work might contain certain biases originated from the human annotations. Yet we believe that the proposed tasks and methods can benefit various event-centric NLP/NLU tasks like event extraction, task-oriented dialogue systems, and so forth.

References

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8, Sydney, Australia. Association for Computational Linguistics.

Emmon Bach. 1986. The algebra of events. *Linguistics and philosophy*, pages 5–16.

Cosmin Bejan and Sanda Harabagiu. 2010. Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1412–1422, Uppsala, Sweden. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. 2018. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia. Association for Computational Linguistics.

Bernard Comrie. 1976. *Aspect: An introduction to the study of verbal aspect and related problems*, volume 2. Cambridge university press.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Annemarie Friedrich and Alexis Palmer. 2014. Situation entity annotation. In *Proceedings of LAW VIII - The 8th Linguistic Annotation Workshop*, pages 149–158, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Annemarie Friedrich, Alexis Palmer, and Manfred Pinkal. 2016. Situation entity types: automatic classification of clause-level aspect. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1757–1768, Berlin, Germany. Association for Computational Linguistics.

William Gantt, Lelia Glass, and Aaron Steven White. 2022. Decomposing and recomposing event structure. *Transactions of the Association for Computational Linguistics*, 10:17–34.

Goran Glavaš and Jan Šnajder. 2014. Constructing coherent event hierarchies from news stories. In *Proceedings of TextGraphs-9: the workshop on Graph-based Methods for Natural Language Processing*, pages 34–38, Doha, Qatar. Association for Computational Linguistics.

Goran Glavaš, Jan Šnajder, Marie-Francine Moens, and Parisa Kordjamshidi. 2014. HiEve: A corpus for extracting event hierarchies from news stories. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3678–3683, Reykjavik, Iceland. European Language Resources Association (ELRA).

Rujun Han, I-Hung Hsu, Jiao Sun, Julia Baylon, Qiang Ning, Dan Roth, and Nanyun Peng. 2021. ESTER: A machine reading comprehension dataset for reasoning about event semantic relations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7543–7559, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paul J Hopper and Sandra A Thompson. 1980. Transitivity in grammar and discourse. *language*, pages 251–299.

Nancy Ide, Collin Baker, Christiane Fellbaum, Charles Fillmore, and Rebecca Passonneau. 2008. MASC: the manually annotated sub-corpus of American English. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Thomas Kober, Malihe Alikhani, Matthew Stone, and Mark Steedman. 2020. Aspectuality across genre: A distributional semantics approach. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4546–4562, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Manfred Krifka. 1989. Nominal reference, temporal constitution and quantification in event semantics. *Semantics and contextual expression*, 75:115.

Tao Li, Vivek Gupta, Maitrey Mehta, and Vivek Srikumar. 2019. A logic-driven framework for consistency of neural 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 3924–3935, Hong Kong, China. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Andrei Mikheev, Claire Grover, and Marc Moens. 1998. Description of the LTG system used for MUC-7. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998.*

George A. Miller. 1992. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992.*

Paramita Mirza and Sara Tonelli. 2014. An analysis of causality between events and its relation to temporal information. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2097–2106, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Marc Moens and Mark Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15–28.

Sean Monahan and Mary Brunson. 2014. Qualities of eventiveness. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 59–67, Baltimore, Maryland, USA. Association for Computational Linguistics.

Sean Monahan, Michael Mohler, Marc T Tomlinson, Amy Book, Maxim Gorelkin, Kevin Crosby, and Mary Brunson. 2015. Populating a knowledge base with information about events. In *TAC*.

Alexander PD Mourelatos. 1978. Events, processes, and states. *Linguistics and philosophy*, 2(3):415–434.

Tim O'Gorman, Kristin Wright-Bettner, and Martha Palmer. 2016. Richer event description: Integrating event coreference with temporal, causal and bridging annotation. In *Proceedings of the 2nd Workshop on Computing News Storylines (CNS 2016)*, pages 47–56, Austin, Texas. Association for Computational Linguistics.

Xingyuan Pan, Maitrey Mehta, and Vivek Srikumar. 2020. Learning constraints for structured prediction using rectifier networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4843–4858, Online. Association for Computational Linguistics.

Xingyuan Pan and Vivek Srikumar. 2016. Expressiveness of rectifier networks. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 2427–2435, New York, New York, USA. PMLR.

James Pustejovsky, José M Castano, Robert Ingria, Roser Sauri, Robert J Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R Radev. 2003. Timeml: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

Zeno Vendler. 1957. Verbs and times. *The philosophical review*, 66(2):143–160.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. 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 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium*.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020a. Joint constrained learning for event-event relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 696–706, Online. Association for Computational Linguistics.

Haoyu Wang, Hongming Zhang, Muhao Chen, and Dan Roth. 2021. Learning constraints and descriptive segmentation for subevent detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5216–5226, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haoyu Wang, Hongming Zhang, Yuqian Deng, Jacob R Gardner, Muhao Chen, and Dan Roth. 2022a. Extracting or guessing? improving faithfulness of event temporal relation extraction. *arXiv preprint arXiv:2210.04992*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020b. MAVEN: A Massive General Domain Event Detection Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1652–1671, Online. Association for Computational Linguistics.

Yiwei Wang, Muhao Chen, Wenxuan Zhou, Yujun Cai, Yuxuan Liang, Dayiheng Liu, Baosong Yang, Juncheng Liu, and Bryan Hooi. 2022b. Should we rely on entity mentions for relation extraction? debiasing relation extraction with counterfactual analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3071–3081, Seattle, United States. Association for Computational Linguistics.

Yadollah Yaghoobzadeh and Hinrich Schütze. 2015. Corpus-level fine-grained entity typing using contextual information. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 715–725, Lisbon, Portugal. Association for Computational Linguistics.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2021. Zero-shot Label-aware Event Trigger and Argument Classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1331–1340, Online. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Ben Zhou, Qiang Ning, Daniel Khashabi, and Dan Roth. 2020. Temporal common sense acquisition with minimal supervision. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7579–7589, Online. Association for Computational Linguistics.

Appendix

	Modality	Affirmation	Specificity	Telicity	Durativity	Kinesis
# of cases	Realis:Irrealis 6327:1072	Affirmative:Negative 6732:667	Specific:Generic 4445:2954	Telic:Atelic 1298:6101	Durative:Punctual 6773:626	Action:State 4278:3121

Table 5: Dataset statistics.

Figure 4: The event property annotation of “acknowledge” in the annotation interface.

Figure 5: The event property annotation of “display” in the annotation interface.

Figure 6: Annotation guideline for durativity and telicity.

Modality

- **Realis**
  - **Context-independent: World War II**
  - **Context-dependent: I hired an assistant who speaks English.**
- **Irrealis**
  - **Context-independent: Imagine**
  - **Context-dependent: I'm looking for an assistant who speaks English.**

This task asks you to annotate the mode of the highlighted verb. You have three choices: realis, irrealis, or uncertain. If you think the highlighted verb is happening in real world, you should choose affirmative; if you think the highlighted verb is fictive or unreal, you should choose irrealis; if you are uncertain, choose uncertain. To help your understanding, you can refer to the following example:

I hired an assistant who speaks English: Realis
I'm looking for an assistant who speaks English: Irrealis

Note: if the sentence is not complete, you can always associate a realistic subject with the verb.

此任务要求您标注动词是否为现实发生的。您有三个选择：现实、非现实或不确定。如果您认为字体加粗的动词是现实发生的，您应该选择 现实；如果您认为该动词不是现实发生的，您应该选择 非现实；如果您不确定，请选择 不确定。为方便您的理解，您可以参考下面这个例子：

我雇了一个说英语的助手：现实
我要雇一个说英语的助手：非现实

Genericity

- **Generic**
  - **Context-independent: World War II**
  - **Context-dependent: I hired an assistant who speaks English.**
- **Specific**
  - **Context-independent: Imagine**
  - **Context-dependent: I'm looking for an assistant who speaks English.**

This task asks you to annotate the genericity of the highlighted verb. You have three choices: generic, specific, or uncertain. If you think the highlighted verb is described in a generic way, you should choose Generic; if you think the highlighted verb is describing a specific case, you should choose specific; if you are uncertain, choose uncertain. To help your understanding, you can refer to the following example:

Lions eat meat: Generic
My boss is looking for an assistant who speaks English: Specific

Note: if the sentence is not complete, you can always associate a realistic subject with the verb.

此任务要求您标注动词是否为具体的。您有三个选择：具体、非具体或不确定。如果您认为字体加粗的动词是具体发生的，您应该选择 具体；如果您认为该动词在描述一个通用的场景，您应该选择 非具体；如果您不确定，请选择 不确定。为方便您的理解，您可以参考下面这个例子：

狮子吃肉：非具体
我的老板要雇一个说英语的助手：具体

Figure 7: Annotation guideline for modality and genericity.

Kinesis

- **State**
  - **Context-independent: Love**
  - **Context-dependent: She is working. Don't interrupt her.**
- **Non-state**
  - **Context-independent: Hug**
  - **Context-dependent: He works out in the gym two or three times a week.**

This task asks you to annotate the kinesis of the highlighted verb. You have three choices: state, non-state, or uncertain. If you think the highlighted verb is describing a state, you should choose state; if you think the highlighted verb describes a non-state, or action, you should choose non-state; if you are uncertain, choose uncertain. To help your understanding, you can refer to the following example:

He loves me: State
He hugs me: Non-state

此任务要求您标注动词的运动性。您有三个选择：状态、非状态或不确定。如果您认为字体加粗的动词描述了一种状态，您应该选择 状态；如果您认为该动词描述了一个动作，您应该选择 非状态；如果您不确定，请选择 不确定。为方便您的理解，您可以参考下面这个例子：

他爱我：状态
他抱住了我：动作

Affirmation

- **Affirmative**
  - **Context-independent: Admit**
  - **Context-dependent: I can't help feeling that …**
- **Negative**
  - **Context-independent: Deny**
  - **Context-dependent: We have no more bread.**

This task asks you to annotate the affirmation of the highlighted verb. You have three choices: affirmative, negative, or uncertain. If you think the highlighted verb is affirmative, you should choose affirmative; if you think the highlighted verb is negative, you should choose negative; if you are uncertain, choose uncertain. To help your understanding, you can refer to the following example:

I can't help feeling that: Affirmative
We have no more bread: Negative

此任务要求您标注动词是否表达了肯定的含义。您有三个选择：肯定、否定或不确定。如果您认为字体加粗的动词表达了一个肯定的含义，您应该选择 肯定；如果您认为该动词表达了一个否定的含义，您应该选择 否定；如果您不确定，请选择 不确定。为方便您的理解，您可以参考下面这个例子：

我不禁感到害怕：肯定
我们没有面包了：否定

Figure 8: Annotation guideline for kinesis and affirmation.