Transitive Consistency Constrained Learning for Entity-to-Entity Stance Detection

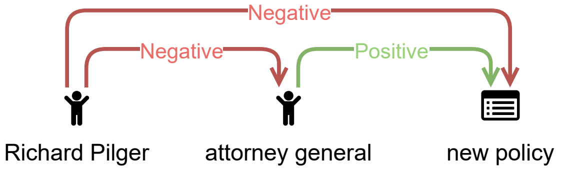
Haoyang Wen, Eduard Hovy, Alexander Hauptmann
[†] Language Technologies Institute, Carnegie Mellon University
[‡] School of Computing and Information Systems, The University of Melbourne
{hwen3, hovy, alex}@cs.cmu.edu

Abstract

Entity-to-entity stance detection identifies the stance between a pair of entities with a directed link that indicates the source, target and polarity. It is a streamlined task without the complex dependency structure for structural sentiment analysis, while it is more informative compared to most previous work assuming that the source is the author. Previous work performs entity-to-entity stance detection training on individual entity pairs. However, stances between inter-connected entity pairs may be correlated. In this paper, we propose transitive consistency constrained learning, which first finds connected entity pairs and their stances, and adds an additional objective to enforce the transitive consistency. We explore consistency training on both classification-based and generation-based models and conduct experiments to compare consistency training with previous work and large language models with in-context learning. Experimental results illustrate that the inter-correlation of stances in political news can be used to improve the entity-to-entity stance detection model, while overly strict consistency enforcement may have a negative impact. In addition, we find that large language models struggle with predicting link direction and neutral labels in this task.[1]

1 Introduction

Detecting polarity from text has been widely studied in different forms, such as sentence-level (Pang et al., 2002) or aspect-level sentiment analysis (Pontiki et al., 2014), target-oriented stance detection (Hu and Liu, 2004; Somasundaran and Wiebe, 2010), and structured analysis (Kim and Hovy, 2004; Wiebe et al., 2005; Barnes et al., 2022). Some recent efforts explore a streamlined and informative form, entity-to-entity stance detection (Park et al., 2021; Zhang et al., 2022), which identifies the stance between a pair of entities with a directed link that indicates source, target, and polarity. Entity-to-entity stance detection can be used to analyze more objective contexts such as news articles in an effective way without the extraction of complex dependency structure with opinion expressions (Kim and Hovy, 2004; Wiebe et al., 2005; Barnes et al., 2022), especially compared to most previous work that usually assumes opinions come from the author (Pang et al., 2002; Hu and Liu, 2004; Somasundaran and Wiebe, 2010; Pontiki et al., 2014; Mohammad et al., 2016).



(e1, Richard Pilger), director of the elections crimes branch in the Justice Department's Public Integrity Section, told colleagues in an email that the **(e2, attorney general)** was issuing "an important **(e3, new policy)** abrogating the forty-year-old Non-Interference Policy for ballot fraud investigations in the period prior to elections becoming certified and uncontested".

Figure 1: An example of three entity-to-entity stances and their consistency. If we know "Richard Pilger" was against the "attorney general", and the "attorney general" supported the "new policy", we may infer that "Richard Pilger" was also likely against the "new policy".

The input of a typical entity-to-entity stance detection system consists of a context and a pair of entities and finds a directed link between them, as shown in Figure 1. Previous efforts (Park et al., 2021; Zhang et al., 2022) optimize model training on each entity pair individually. However, the stances of inter-connected entity pairs may be correlated. As we can find in Figure 1, if we know "Richard Pilger" was against the "attorney general" (Negative), while the attorney general supported the "new policy" (Positive), we may infer that

[1] Our code is available at https://github.com/wenhycs/ACL-2024-Transitive-Consistency-Constrained-Learning-for-Entity-to-Entity-Stance-Detection.

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1467–1480
August 11-16, 2024 ©2024 Association for Computational Linguistics

"Richard Pilger" was also against the new policy with these two known stances. We hypothesize that this type of transitive correlation is common in political news and can be used effectively to train better models (Sobhani et al., 2017).

In this paper, we consider the correlation between inter-connected entity pairs as transitive consistency constraints during training and use these constraints to help learn entity-to-entity stance detection models. Specifically, we first sample a pair of sentences that share a common entity. Based on the intra-sentence entity-to-entity stances, we may infer the stance of the entity pair across the two sentences with transitivity. The inferred stance is expected to be softly aligned with the stance detection prediction on the entity pair directly. Therefore, given the two intra-sentence stance predictions and the cross-sentence stance prediction, we can add additional soft consistency loss between the triple terms to enforce the similarity. In this work, we develop two typical methods for entity-to-entity stance detection and try to combine the transitive consistency constraints with them. One is based on relation classification from entity pair representations (Eberts and Ulges, 2020; Wang and Lu, 2020; Wang et al., 2020). The other method is based on recent trends in language model instruction tuning (Wei et al., 2022; Sanh et al., 2022; Chung et al., 2022; Muennighoff et al., 2023), where we generate the stance autoregressively.

We conduct our experiments on DSE (Park et al., 2021) and SEESAW (Zhang et al., 2022), both of which analyze stances in political news. DSE requires models to identify the neutral label, and the label direction. SEESAW is originally designed to jointly generate an entity pair and the corresponding polarity, so it does not provide the mention-level entity annotation and the neutral label. Our experiment results show that the transitive consistency constraints help in learning better classification and generation models, which also implies the prevalence of stance transitivity on political news. We further show that the performance is sensitive to the degree of applying constraints, and there is a performance degradation if we overstrictly enforce the constraints. In addition, we find that large language models with in-context learning (Muennighoff et al., 2023; Touvron et al., 2023) cannot obtain reliable performance on DSE. Our further analysis shows that it is non-trivial to directly use large language models with in-context learning on the neutral label or directed label predictions.

2 Entity-to-Entity Stance Detection Frameworks

Entity-to-entity stance detection identifies the stance between a pair of entities, as well as the source and target through a directed link. In this section, we introduce two basic frameworks for this challenge. One is based on relation classification using entity pair representations, while the other is to generate the stance autoregressively.

2.1 Classification-Based Framework

Classification-based framework obtains entity-pair representation from the sentence input and performs classification using the obtained pair-wise representation. This paradigm has shown effectiveness in various relation extraction tasks (Eberts and Ulges, 2020; Wang and Lu, 2020; Wang et al., 2020; Wen and Ji, 2021).

Specifically, the model takes a sequence of tokens $\boldsymbol{x}$ with length n as input, representing the input sentence. The input also includes the positions of two entity mentions (e_1, e_2) in the text. We denote the groundtruth stance as $s(e_1, e_2)$. In this task, we only consider the position of the first token from the corresponding entity, and we denote the positions of the entity pair (e_1, e_2) as (p_1, p_2).

The entity-to-entity stance detection model predicts the stance between the given two entity mentions. It first uses a pretrained language model (Devlin et al., 2019; Liu et al., 2019, PLM) to obtain the contextualized representation of the input sequence,

$$\boldsymbol{H} = \text{PLM}(\boldsymbol{x}),$$

where $\boldsymbol{H}$ represents the contextualized representation of the sequence, and $\boldsymbol{h}_i$ is the representation for the token at position i.

We obtain the entity-pair representation by concatenating the representation of the given position pair

$$\boldsymbol{c} = [\boldsymbol{h}_{p_1}; \boldsymbol{h}_{p_2}] \,.$$

Then entity-pair representation is used for classification with a two-layer feed-forward neural network (FFN) and a softmax layer to predict the entity-to-entity stance label s

$$\begin{aligned} p\left(s \mid e_1, e_2\right) &= \text{softmax}\left(\boldsymbol{a}\right), \\ \boldsymbol{a} &= \text{FFN}_2(\tanh(\text{FFN}_1(\boldsymbol{c}))), \end{aligned}$$

where $\text{FFN}_i(\boldsymbol{h}) = \boldsymbol{W}_i\boldsymbol{h} + \boldsymbol{b}_i$.

For stance detection task that requires models to detect both the polarity and direction (Park et al.,

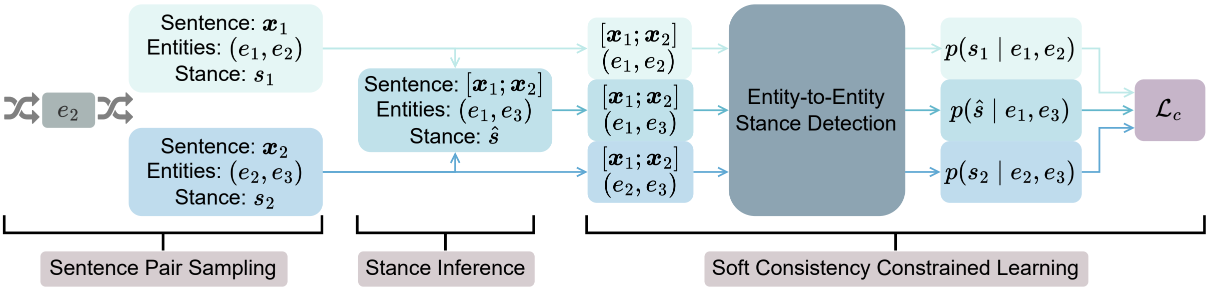

Sentence: $\boldsymbol{x}_1$
Entities: (e_1, e_2)
Stance: s_1
e_2
Sentence: $\boldsymbol{x}_2$
Entities: (e_2, e_3)
Stance: s_2
Sentence: $[\boldsymbol{x}_1; \boldsymbol{x}_2]$
Entities: (e_1, e_3)
Stance: $\hat{s}$
$[\boldsymbol{x}_1; \boldsymbol{x}_2]$ (e_1, e_2)
$[\boldsymbol{x}_1; \boldsymbol{x}_2]$ (e_1, e_3)
$[\boldsymbol{x}_1; \boldsymbol{x}_2]$ (e_2, e_3)
Entity-to-Entity Stance Detection
$p(s_1 \mid e_1, e_2)$
$p(\hat{s} \mid e_1, e_3)$
$p(s_2 \mid e_2, e_3)$
$\mathcal{L}_c$
Sentence Pair Sampling
Stance Inference
Soft Consistency Constrained Learning

Figure 2: The overall framework of soft consistency constrained learning objective. We first sample an entity as the shared entity and use this entity to sample two sentences that can be used for stance inference. We then concatenate the two sentences, with combinations of entity pairs from the sampled three entities for entity-to-entity stance detection, and the objective is the penalty for inconsistent predictions.

2021), each classification label is the combination of the direction and polarity. Therefore, the stance label is related to the input order of the entity pair representation. For example, the label can be "`Entity 1 to Entity 2 positive`" or "`Entity 2 to Entity 1 negative`". "`Entity 1`" represents the first entity of the concatenated entity pair representation while "`Entity 2`" represents the second entity. Therefore, "`Entity 1 to Entity 2`" indicates that the first entity is the source entity while the second entity is the target entity. The only exception is the neutral label, which is undirected in nature, representing that there is no explicit stance polarity between the two entities.

The model is trained by minimizing the cross-entropy loss

$$\mathcal{L}_s = -\sum \mathbb{I}_{s(e_1,e_2)=s_i} \log p(s = s_i \mid e_1, e_2).$$

2.2 Generation-Based Framework

Generation-based methods have also shown strong performance on various tasks, especially for tasks that are not traditionally modeled with generative methods (Lewis and Fan, 2019; Yan et al., 2021; Li et al., 2021; Raffel et al., 2022; Wen et al., 2023). Recently, a line of research utilizes conditional language models to perform relation extraction and achieves promising performance (Paolini et al., 2021; Huguet Cabot and Navigli, 2021; Lu et al., 2022; Wadhwa et al., 2023). Therefore, we also use the generation-based method on our entity-to-entity stance detection experiments.

Specifically, our generation-based model is trained on decoder-only language models (Radford et al., 2019; Muennighoff et al., 2023), which takes input tokens and generates new tokens autoregressively using one Transformer (Vaswani et al., 2017)

$$p(\boldsymbol{o} \mid \boldsymbol{x}, e_1, e_2) = \prod_{i=1}^{|\boldsymbol{o}|} p(o_i \mid \boldsymbol{o}_{<i}; \mathrm{T}(\boldsymbol{x}, e_1, e_2)),$$

where $\boldsymbol{x}$ is the input sentence, and $\mathrm{T}(\boldsymbol{x}, e_1, e_2)$ produces a combination of short instruction, input sentence, and entity pairs into a single sequence with a template. In our entity-to-entity stance detection task, we define the template as: "`Analyze the entity-entity stance in the following text:\n`$\boldsymbol{x}$`\nEntity 1: `e_1`\nEntity 2: `e_2`\nStance:`".

The model takes $\mathrm{T}(\boldsymbol{x}, e_1, e_2)$ and produces a series of tokens $\boldsymbol{o}$ as the output of the entity-to-entity stance detection. Similar to the classification-based method, we need to combine direction and polarity into the output of the generation when performing a directed stance detection. We first output the stance direction, and then output the stance polarity. We use text `Entity 1 to Entity 2` and `Entity 2 to Entity 1` to represent two directions, and `positive`, `negative`, and `neutral` as polarity words. The output text of a neutral label does not include a direction phrase as it is undirected.

The model is trained by minimizing the log-likelihood over the generated output sequence:

$$\begin{aligned}\mathcal{L}_s &= -\log p(\boldsymbol{o} \mid \boldsymbol{x}, e_1, e_2) \\ &= -\sum_{i=1}^{|\boldsymbol{O}|} \log p(o_i \mid \boldsymbol{o}_{<i}; \mathrm{T}(\boldsymbol{x}, e_1, e_2)).\end{aligned}$$

3 Transitive Consistency Constrained Learning

Inter-connected stances may be correlated, especially in political news, as shown in Figure 1. We hope to capture the correlation from optimizing the

		$e_2 \longrightarrow e_3$		$e_2 \longleftarrow e_3$	
		Positive	Negative	Positive	Negative
$e_1 \longrightarrow e_2$	Positive	$e_1 \xrightarrow{\text{Positive}} e_3$	$e_1 \xrightarrow{\text{Negative}} e_3$	-	-
	Negative	$e_1 \xrightarrow{\text{Negative}} e_3$	$e_1 \xrightarrow{\text{Positive}} e_3$	-	-
$e_1 \longleftarrow e_2$	Positive	-	-	$e_1 \xleftarrow{\text{Positive}} e_3$	$e_1 \xleftarrow{\text{Negative}} e_3$
	Negative	-	-	$e_1 \xleftarrow{\text{Negative}} e_3$	$e_1 \xleftarrow{\text{Positive}} e_3$

Table 1: The transitive mapping of a pair of directed stances with a shared entity. "-" denotes no mapping between the pair of stances. We also do not apply transitive mapping for neutral samples.

predicted stances that can be inferred from the transitivity of existing stances. In this section, we will introduce the concept of transitive stance inference.

The transitive stance inference requires multiple inter-correlated entity pairs in a context, while most existing resources only annotate one entity-to-entity stance at the sentence level. Therefore, we propose a simple sentence pair sampling method that helps obtain data for transitive inference. Then we introduce the constrained learning method, which can be added to both classification-based and generation-based methods to capture the transitive correlation. The overall framework is illustrated in Figure 2.

3.1 Transitive Stance Inference

Suppose we have three entities (e_1, e_2, e_3), and we know the directed entity-to-entity stance $s(e_1, e_2)$, $s(e_2, e_3)$, the stance inference is to infer the stance from the two known stances

$$\hat{s}(e_1, e_3) = s(e_1, e_2) \oplus s(e_2, e_3).$$

The stance inference can be divided into two steps. The first step is to check whether e_1 can reach e_3 ($e_1 \to e_2 \to e_3$) or e_3 can reach e_1 ($e_1 \leftarrow e_2 \leftarrow e_3$) using the existing directed links, which is a prerequisite for transitivity. If e_1 can reach e_3, we will be able to infer the stance from e_1 (as the source) towards e_3 (as the target), and vice versa. For other cases ($e_1 \to e_2 \leftarrow e_3$, $e_1 \leftarrow e_2 \to e_3$), we will not be able to apply the transitive inference. We also do not use stances with neutral labels in our stance inference as they are undirected.

The second step is to determine the stance polarity. We formulate the stance polarity mapping similar to the logical non-equivalence (XOR) operator and we denote the mapping operator by $\oplus$. If both polarities of the two known stances $s(e_1, e_2)$, $s(e_2, e_3)$ are positive or negative, the inferred polarity of the stance $\hat{s}(e_1, e_3)$ will be positive. If among the two known stances, one is positive and the other is negative, the inferred polarity of the stance $\hat{s}(e_1, e_3)$ will be negative.

Combining these two steps, we have a complete stance inference from transitive mapping, which is illustrated in Table 1.

3.2 Two-Step Sentence Pair Sampling

Existing resources (e.g., Park et al., 2021) mostly focus on sentence-level annotation. For each sentence, they pick one pair of entities and annotate the directed stance between them. However, as we introduced in Section 3.1, to infer stance with transitivity, we will need a pair of stances of which two entity pairs share one entity and there are in total three entities. Therefore, we propose a simple sentence pair sampling method in the training data using a two-step sampling to obtain these samples.

Specifically, we first uniformly sample an entity as the shared entity. Uniform sampling over entities is to ensure that a few frequently occurring entities will not have a substantially high probability of being sampled. Then, we can find all sentences with entity-to-entity stance annotations involving the given entity, and we uniformly sample a pair of sentences among them. The sentence pair will also provide us with a pair of entity-to-entity stance annotations that share a common entity. We will disregard the sampled sentence pair if the entity pairs from the sentence pair are the same, or if the sampled entity-to-entity stance pair does not constitute the case we can apply the transitive mapping. We keep performing the two-step sampling until we find a valid sentence pair.

3.3 Soft Consistency Constrained Learning

The overall idea of constrained learning is to add an additional penalty if the predicted label does not match the inferred label (Wang et al., 2020). We use the classification-based method to explain our proposed method first, and naturally extend it to the generation-based method.

For a given sampled sentence pair $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ with entity pair $(e_{1,1}, e_{1,2})$ and $(e_{2,1}, e_{2,2})$ correspondingly, we perform normalization on the sentence pair annotation first to ensure $e_{1,2} = e_{2,1}$ as the shared entity. This normalization involves flipping the stance direction with the entity order within the input entity pair. For example, if there is a stance $s(e_1, e_2)$ interpreted as `Entity 1 to Entity 2 positive`, after flipping the input order of the entity pair from (e_1, e_2) to (e_2, e_1), the corresponding flipped stance $s(e_2, e_1)$ will be `Entity 2 to Entity 1 positive`. Therefore, we flip the label of the first sentence if the shared entity is $e_{1,1}$ in the original annotation, and flip the label of the second sentence if the shared entity is $e_{2,2}$. For simplicity, we assume that the input of the following discussion is already normalized.

We use concatenated input from the sentence pair $[\boldsymbol{x}_1; \boldsymbol{x}_2]$ with three entity pairs $(e_{1,1}, e_{1,2})$, $(e_{2,1}, e_{2,2})$ and $(e_{1,1}, e_{2,2})$, which represents two intra-sentence entity pairs with groundtruth stance annotation, and one inter-sentence entity pair with inferred stance. These inputs will be fed into the classification-based method and obtain three distributions, $p(s \mid e_{1,1}, e_{1,2})$, $p(s \mid e_{2,1}, e_{2,2})$ and $p(s \mid e_{1,1}, e_{2,2})$. The objective is to promote similarity between $p(s \mid e_{1,1}, e_{1,2}) \times p(s \mid e_{2,1}, e_{2,2})$ and $p(s \mid e_{1,1}, e_{2,2})$, where the former term can be considered as the probability of applying stance inference, while the latter term is the probability of direct stance detection. We use the groundtruth and inferred labels with L_1 distance as this objective

$$\begin{aligned}\mathcal{L}_c = &\,|\log p\left(s = s\left(e_{1,1}, e_{1,2}\right) \mid e_{1,1}, e_{1,2}\right)\\ &+ \log p\left(s = s\left(e_{2,1}, e_{2,2}\right) \mid e_{2,1}, e_{2,2}\right)\\ &- \log p\left(s = \hat{s}\left(e_{1,1}, e_{2,2}\right) \mid e_{1,1}, e_{2,2}\right)| \quad .\end{aligned}$$

We jointly train the consistency constrained objective with the regular single-sentence learning objective $\mathcal{L}_s$ (cross-entropy for classification and sequence log-likelihood for generation)

$$\mathcal{L} = \mathcal{L}_s + \lambda \mathcal{L}_c,$$

where the factor λ is to control the degree of enforcing the consistency objective.

Extending to generation-based method. When extending the consistency constrained learning to generation-based method, we need to find a legitimate estimate from the generation framework to represent the log probability of the stance label. For simplicity, we directly choose the sum of the log probability of the predicted polarity word and two entity numbers to represent the log probability, as they are the most important factors of an entity-to-entity stance label.

Category	DSE	SEESAW
# Label Types	5	2
Stance Direction	In Labels	Part of Input
Neutral Label	Yes	No
Entity Position	Yes	No
Data Statistics		
# Train	13,144	6,263
# Valid	1,461	2,436
# Test	1,623	1,920

Table 2: Comparison of DSE and SEESAW datasets.

4 Experiments

4.1 Data

We conduct experiments on two datasets, DSE (Park et al., 2021) and SEESAW (Zhang et al., 2022). DSE requires the model to predict both the stance direction and the polarity with entity mentions and their positions in the context. The annotation is always from the first mentioned entity to the second entity in the context.

SEESAW was originally designed to jointly generate pairs of entities with their stances, and they do not provide mention-level entities and neutral labels. Instead, all the entities are in canonical form without positions in the context. We slightly change the original experiment setup to make it more consistent with the DSE setting, providing the entity pair with the stance direction as part of the input. In this setting, the models are only asked to detect the non-neutral polarity, given an entity pair and the stance direction.

As a result, for experiments on DSE, we can naturally use both methods introduced in this paper. While on SEESAW, the pair-wised classification method is replaced with sentence-level classification, which uses the name and the direction of the entity pair and context in a question-answering-based pair input. The detailed statistics and comparison of the two datasets are provided in Table 2.

Methods	Development Set		Test Set	
	Micro F_1	Macro F_1	Micro F_1	Macro F_1
LNZ (Combined)	69.40	65.16	70.55	53.58
LNZ (Context)	63.31	45.18	63.71	46.65
LNZ (EntityPrior)	59.14	44.27	58.53	40.63
DSE2QA (Complete)	78.92	67.51	77.26	66.17
DSE2QA (Pseudo)	80.72	68.27	79.73	67.66
POLITICS	85.45	71.94	84.19	71.12
Generation	83.92	70.14	83.25	70.12
+ Consistency Training	84.86	71.75	83.51	70.25
Classification	85.82	74.07	83.82	70.59
+ Consistency Training	**86.67**	**74.41**	**85.19**	**72.50**
BLOOMZ-176b + 25 samples	20.60	18.04	21.07	18.56

Table 3: Results on DSE dataset. The performances of our methods are averaged performance (%) over 5 runs.

Methods	Mirco F_1
DSE2QA	83.35
POLITICS	84.02
Generation	80.35
+ Consistency Training	81.05
Classification	83.72
+ Consistency Training	**84.11**
BLOOMZ-176b + 10 samples	77.29

Table 4: Results on SEESAW dataset. Different from the original setting of SEESAW, we provide entity pair and direction as the input and ask models to predict non-neutral stance. The performances we reported are averaged performance (%) over 5 runs.

4.2 Experimental Setup

For the classification-based method, we use RoBERTa (Liu et al., 2019) as the pretrained language model to obtain the entity pair representations and we choose `roberta-base`[2] as the base checkpoint to initialize the model. For the generation-based method, we finetune an openly available instruction-tuned large language model series, BLOOMZ (Muennighoff et al., 2023). We use `bloomz-560m`[3] as the initial checkpoint as the model size is close to RoBERTa. More details of experimental setup and computational infrastructures can be found in Appendix A.

Comparing with previous work. We compare our work with some previous work, including LNZ (Liang et al., 2019), DSE2QA (Park et al., 2021) and POLITICS (Liu et al., 2022) on DSE. LNZ is a pairwise classification model combining the entity prior representation and entity representation in the context. DSE2QA converts the stance detection problem into a series of template-based question answering. POLITICS is a pretrained model with ideological information.

As we alter the original experimental setting of SEESAW, we provide our own implementation of DSE2QA and POLITICS in this data and compare it against our method. Especially, on these two datasets, we apply POLITICS with the same classification framework as our model, the only difference is their ideology-aware pretrained model.

In addition, we also compare our method with large language model from the same series as our generation model, BLOOMZ-176b (Muennighoff et al., 2023)[4] with few-shot in-context learning samples on both datasets, to understand the capability of existing large language models in performing this entity-to-entity stance detection task. We take 5-shot samples of each label (in total 25 samples in DSE and 10 samples in SEESAW) to perform the language model inference. Additional results with Llama2-70b-chat (Touvron et al., 2023) on DSE are provided in Appendix B.

4.3 Results

Table 3 shows the experimental results on the DSE dataset, where we can find steady improvement from adding the transitive consistency constrained learning to the classification-based (Gen-

[2] https://huggingface.co/roberta-base

[3] https://huggingface.co/bigscience/bloomz-560m

[4] https://huggingface.co/bigscience/bloomz

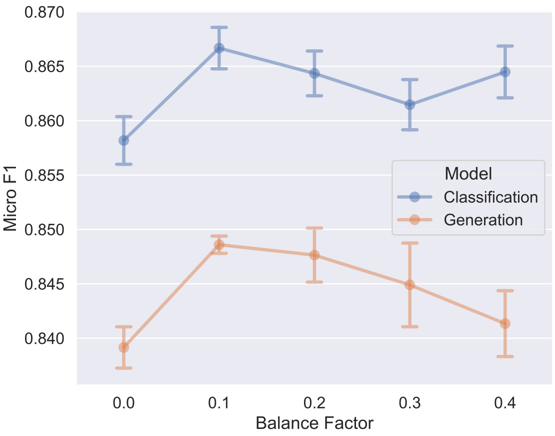


Figure 3: The Micro F_1 performances of generation and classification models on DSE development set with different balance factor λ. In general, we can observe that with the increase of λ, the performance first improves and then degrades.

eration + Consistency Training) and generation-based method (Generation + Consistency Training). We also observe that the improvement for generation-based method method is smaller than classification-based method, indicating that there is still room to further investigate better methods to incorporate constrained learning into generative modeling. The results of a large language model with in-context learning is illustrated with BLOOMZ-176b. This result indicates that BLOOMZ-176b has a deficient performance on DSE, the entity-to-entity stance detection task, and few-shot in-context learning cannot substantially help with learning well on this task.

On SEESAW dataset, we also find that the consistency constrained learning provides consistent improvement to two base methods. The performance of classification-based consistency constrained method outperforms or is on par with previous work, specifically compared to POLITICS, the model pretrained with ideology information. We can also observe that the absolute improvement is slightly less than what we observe in DSE, which indicates that the constrained learning objective works better when stance directions are part of the prediction output. BLOOMZ-176b, contrary to the DSE results, also provides fair performance on this dataset. We will further discuss the large language model performance discrepancy between DSE and SEESAW in Section 4.6.

Training	Micro F_1	Macro F_1
Data Augmentation	84.55	71.91
Consistency Learning	**85.19**	**72.50**

Table 5: Comparison between vanilla data augmentation and soft consistency constrained learning. Both methods use the same two-step sampling method to obtain the inferred stance of the cross-sentence entity pair.

Sampling	Micro F_1	Macro F_1
Random Sampling	84.44	71.99
Two-Step Sampling	**85.19**	**72.50**

Table 6: Effects of two-step sampling method compared to the vanilla uniform random sampling over all valid sentence pairs for stance inference on DSE test set.

4.4 Effects of Soft Consistency Constrained Learning

We further analyze the effects of the soft consistency constrained learning and illustrate the results in Figure 3 on the DSE dataset. We can observe that after involving the soft consistency constrained objective, the performances compared to the one without constrained objective ($\lambda = 0$) improve. However, enforcing this objective with large λ, similar to vanilla data augmentation, does not further contribute to the performance, but instead results in performance degradation. This phenomenon suggests that transitivity does not always hold and there is still a chance that the inference is not correct. Therefore, consistency constrained learning requires a carefully chosen soft setup.

In addition, we conduct another experiment, to analyze the performance of soft consistency constrained learning compared to vanilla data augmentation on classification-based method. The vanilla data augmentation uses the sample two-step sampling to obtain the inferred label for the cross-sentence entity pair. The results are shown in Table 5. We can find that both data augmentation and consistency learning can contribute to the model performance, while consistency constraints provide additional performance improvement from learning to make consistent predictions in a context.

4.5 Effects of Two-Step Sampling

We also analyze the effects of two-step sampling. The results are shown in Table 6. We compare the two-step sampling to uniform random sampling

Test Data Type	Micro F_1	Macro F_1
Full Label	21.07	18.56
- w/o Direction	26.74	26.19
- w/o Neutral	66.96	35.80
- w/o Both	77.62	71.73

Table 7: Analysis of BLOOMZ-176b performances on different test data on DSE, including test labels that do not require predicting direction, data excluding neutral samples, and test labels without both of them. The results show that the performance suffers from predicting the neutral labels and directional information.

over all valid sentence pairs for stance inference. The results show that two-step sampling outperforms uniform sampling, indicating that it is important to consider the entity distributions when selecting the sentence pair. Vanilla uniform random sampling over all valid sentence pairs results in a long-tail distribution of the shared entity. While constrained learning performs better when the shared entity follows a uniform distribution.

4.6 Challenge of Large Language Models for Entity-to-Entity Stance Detection

From Table 3, we find surprisingly low performances from the large language model, while the performance on Table 4 is more promising. As we explained in Table 2, the main difference between the two datasets is the requirement of label direction and neutral sample detection. Therefore, we conduct further analysis to understand the performance discrepancy. Besides the original test label, we use test data without label direction (only requiring polarity prediction), test data without neutral label samples, and test data without both factors, to analyze the impact of these factors. We conduct a similar in-context learning scheme as introduced in Section 4.2. The results show that the large language model achieves better performance by removing the requirement of neutral label prediction or predicting directed information. Similar results on Llama2-70b-chat can be found in Appendix B.

This phenomenon reveals that the semantic information of directed stances and neutral stances is not well pretrained in the current large language models. In addition, it is also non-trivial to use the in-context learning method to help large language models obtain the ability to conduct directed stance and neutral stance detection. These results also partly align with Zhang et al. (2023) which shows that large language models lag behind in complex or structured sentiment analysis tasks.

5 Related Work

Earlier efforts on stance detection primarily focus on some specific targets with rich training and testing data (Somasundaran and Wiebe, 2010; Augenstein et al., 2016; Mohammad et al., 2016). A typical model in this setting is built for each target separately (Mohammad et al., 2016; Mohtarami et al., 2018; Siddiqua et al., 2018; Aldayel and Magdy, 2019; Graells-Garrido et al., 2020), or cross-target stance detection, where we have pre-defined leave-out targets to test the model generalization to targets that do not have training data (Xu et al., 2018; Liang et al., 2021; Allaway et al., 2021; Jiang et al., 2022). More recent efforts also study zero-shot or few-shot stance detection on a large number of targets (Allaway and McKeown, 2020; Lin et al., 2021; Liu et al., 2021; He et al., 2022; Liang et al., 2022a,b; Wen and Hauptmann, 2023). This setting requires model to generalize to a large number of unseen targets. Recently, another line of research studies a more objective form of stance detection, entity-to-entity stance detection (Park et al., 2021; Zhang et al., 2022), where we analyze the stance from one entity to another entity in the text. Our work follows this direction and studies the consistency between related entity-to-entity stances and uses this consistency to help model training, compared to previous work (Park et al., 2021; Zhang et al., 2022) that tackles the stances individually.

On the other hand, stance detection can be considered as a simplified task of structured sentiment analysis (Barnes et al., 2022), which identifies opinion holders, targets, expressions, and polarities into dependency structures. Typical stance detection setups assume that the opinions are from the author, and models only need to consider the target. While entity-to-entity stance detection combines holders, targets, and polarities with more streamlined, directed link labels.

The consistency assumption between related stances is also related to multi-target stance detection (Sobhani et al., 2017). Multi-target stance detection is to detect a stance pair for a multi-target (*e.g.,* a pair of targets), assuming that when expressing the stance to one target, it also implies stances to a related target. Similar consistency constraints have also been discussed on polarity link prediction in social networks (Leskovec et al., 2010) and event relation extraction (Wang et al., 2020). The focus of Leskovec et al. (2010) is the network-based link prediction, which is quite different from the text-based analysis. Wang et al. (2020) performs event-event relation extraction (temporal and hierarchical). The document-level annotation provides consistent labels and therefore they do not have steps that we have to create data for consistency training.

6 Conclusion

In this paper, we present a method that models the transitive consistency constraints during training to help train entity-to-entity stance detection models. Our proposed methods first sample sentence pair to conduct stance transitivity inference, and model the constraints as the similarity between the inferred and directly predicted stance. Experiments show that this constrained learning helps improve both classification- and generation-based models. Further analysis indicates that the constrained learning is sensitive to the balance factor which controls the enforcement of constraints during training. We also find large language models may not perform reliable complex structured predictions, especially on neutral and directed samples.

Limitations

In this work, our experimental setup assumes that the entities involved in a context are pre-extracted and we use gold standard entities for stance detection. However, to conduct end-to-end entity-to-entity stance detection, we need an additional prerequisite component for entity extraction, which is not used and covered in this paper. Therefore, it is difficult to compare this work with other work that conducts end-to-end entity-to-entity stance detection or structured sentiment analysis, such as generative entity-to-entity stance detection that jointly finds entities with their stances (Zhang et al., 2022).

Additionally, the consistency constraints in this paper are used during training. However, for large language models discussed in this paper, it is infeasible to conduct full fine-tuning with limited computational resources. It is still under exploration how to use frozen large language models to obtain reliable performances for this challenge, and whether those consistency constraints can also be effectively used in this setup.

In this paper, our experiments are conducted in a specialized domain, political news, in which we generally see more frequent polarized opinions. It is still under exploration whether the stance transitivity constraints widely exist in other domains. If not, we need to find scenarios where transitivity constraints hold and conduct constrained learning on these scenarios specifically. Besides, in general domain, the negation of a positive stance may not be exactly the opposite one, which should also be considered when extending this work to a more general domain. It is also an interesting direction to study the similar transitivity in other settings (e.g. relations in knowledge graphs, semantic concept inheritance).

Acknowledgement

This work used the Delta system at the National Center for Supercomputing Applications through allocation CIS240015 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

Abeer Aldayel and Walid Magdy. 2019. Your stance is exposed! analysing possible factors for stance detection on social media. *Proc. ACM Hum.-Comput. Interact.*, 3(CSCW).

Emily Allaway and Kathleen McKeown. 2020. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8913–8931, Online. Association for Computational Linguistics.

Emily Allaway, Malavika Srikanth, and Kathleen McKeown. 2021. Adversarial learning for zero-shot stance detection on social media. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4756–4767, Online. Association for Computational Linguistics.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas. Association for Computational Linguistics.

Jeremy Barnes, Laura Oberlaender, Enrica Troiano, Andrey Kutuzov, Jan Buchmann, Rodrigo Agerri, Lilja Øvrelid, and Erik Velldal. 2022. SemEval 2022 task 10: Structured sentiment analysis. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1280–1295, Seattle, United States. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *CoRR*, abs/2305.14314.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Markus Eberts and Adrian Ulges. 2020. Span-based joint entity and relation extraction with transformer pre-training. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2006–2013. IOS Press.

Eduardo Graells-Garrido, Ricardo Baeza-Yates, and Mounia Lalmas. 2020. Representativeness of abortion legislation debate on twitter: A case study in argentina and chile. In *Companion Proceedings of the Web Conference 2020*, WWW '20, page 765–774, New York, NY, USA. Association for Computing Machinery.

Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. 2022. Accelerate: Training and inference at scale made simple, efficient and adaptable. `https://github.com/huggingface/accelerate`.

Zihao He, Negar Mokhberian, and Kristina Lerman. 2022. Infusing knowledge from Wikipedia to enhance stance detection. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 71–77, Dublin, Ireland. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA. Association for Computing Machinery.

Pere-Lluís Huguet Cabot and Roberto Navigli. 2021. REBEL: Relation extraction by end-to-end language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2370–2381, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yan Jiang, Jinhua Gao, Huawei Shen, and Xueqi Cheng. 2022. Few-shot stance detection via target-aware prompt distillation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 837–847, New York, NY, USA. Association for Computing Machinery.

Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1367–1373, Geneva, Switzerland. COLING.

Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting positive and negative links in online social networks. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 641–650, New York, NY, USA. Association for Computing Machinery.

Mike Lewis and Angela Fan. 2019. Generative question answering: Learning to answer the whole question. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online. Association for Computational Linguistics.

Bin Liang, Zixiao Chen, Lin Gui, Yulan He, Min Yang, and Ruifeng Xu. 2022a. Zero-shot stance detection via contrastive learning. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 2738–2747, New York, NY, USA. Association for Computing Machinery.

Bin Liang, Yonghao Fu, Lin Gui, Min Yang, Jiachen Du, Yulan He, and Ruifeng Xu. 2021. Target-adaptive graph for cross-target stance detection. In *Proceedings of the Web Conference 2021*, WWW '21, page 3453–3464, New York, NY, USA. Association for Computing Machinery.

Bin Liang, Qinglin Zhu, Xiang Li, Min Yang, Lin Gui, Yulan He, and Ruifeng Xu. 2022b. JointCL: A joint contrastive learning framework for zero-shot stance detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 81–91, Dublin, Ireland. Association for Computational Linguistics.

Shuailong Liang, Olivia Nicol, and Yue Zhang. 2019. Who blames whom in a crisis? detecting blame ties from news articles using neural networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 655–662. AAAI Press.

Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. 2021. BertGCN: Transductive text classification by combining GNN and BERT. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1456–1462, Online. Association for Computational Linguistics.

Rui Liu, Zheng Lin, Yutong Tan, and Weiping Wang. 2021. Enhancing zero-shot and few-shot stance detection with commonsense knowledge graph. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3152–3157, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Yujian Liu, Xinliang Frederick Zhang, David Wegsman, Nicholas Beauchamp, and Lu Wang. 2022. POLITICS: Pretraining with same-story article comparison for ideology prediction and stance detection. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1354–1374, Seattle, United States. Association for Computational Linguistics.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California. Association for Computational Linguistics.

Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. 2018. Automatic stance detection using end-to-end memory networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 767–776, New Orleans, Louisiana. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Kunwoo Park, Zhufeng Pan, and Jungseock Joo. 2021. Who blames or endorses whom? entity-to-entity directed sentiment extraction in news text. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4091–4102, Online. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2022. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Umme Aymun Siddiqua, Abu Nowshed Chy, and Masaki Aono. 2018. Stance detection on microblog focusing on syntactic tree representation. In *Data Mining and Big Data*, pages 478–490, Cham. Springer International Publishing.

Parinaz Sobhani, Diana Inkpen, and Xiaodan Zhu. 2017. A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 551–557, Valencia, Spain. Association for Computational Linguistics.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124, Los Angeles, CA. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020. 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.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721, Online. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Haoyang Wen and Alexander Hauptmann. 2023. Zero-shot and few-shot stance detection on varied topics via conditional generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1491–1499, Toronto, Canada. Association for Computational Linguistics.

Haoyang Wen and Heng Ji. 2021. Utilizing relative event time to enhance event-event temporal relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10431–10437, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haoyang Wen, Zhenxin Xiao, Eduard Hovy, and Alexander Hauptmann. 2023. Towards open-domain Twitter user profile inference. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3172–3188, Toronto, Canada. Association for Computational Linguistics.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Lang. Resour. Evaluation*, 39(2-3):165–210.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Chang Xu, Cécile Paris, Surya Nepal, and Ross Sparks. 2018. Cross-target stance classification with self-attention networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 778–783, Melbourne, Australia. Association for Computational Linguistics.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2416–2429, Online. Association for Computational Linguistics.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023. Sentiment analysis in the era of large language models: A reality check. *CoRR*, abs/2305.15005.

Xinliang Frederick Zhang, Nick Beauchamp, and Lu Wang. 2022. Generative entity-to-entity stance detection with knowledge graph augmentation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9950–9969, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Detailed Experimental Setup

On DSE, we train the classification-based method using a learning rate of 2e-5. The batch size is 32, and λ is 0.1. We train the model with 30 epochs and evaluate it on the validation set to select the checkpoint with the best validation set performance. For the generation-based method, we use a learning rate of 2e-5. The batch size is 32 for the cross-entropy learning objective and 16 for soft consistency constrained learning. λ is 0.1. We train the generation-based method with 10 epochs, and use the final checkpoint for validation and test set evaluation.

On SEESAW, the generation-based method is trained with a learning rate of 1e-5, batch size of 32 for sequence log-likelihood objective, 16 for soft consistency constrained learning, and λ of 0.3. We train the generation-based method on with 10 epochs, and use the final checkpoint for validation and test set evaluation. The classification-based model, as we mention in Section 4.1, is a sequence classification given the entity pair with the direction, and the context in a question-answering-based sentence pair input. The template for input sentence pair is "`Source Entity:` e_1, `Target Entity:` e_2 `</s>` $\boldsymbol{X}$". We train this classification method using a learning rate of 2e-5. The batch size is 32, and λ is 1.0. We train the model with 30 epochs and evaluate it on the validation set to select the checkpoint with the best validation set performance.

We train all models with a linear scheduler with a warmup rate 0.1. We use FP16 mixed precision training for the generation-based method. We use full fine-tuning on both classification-based and generation-based methods. For sentence pair sampling to conduct stance inference on the SEESAW dataset, we drop all sentences that include special entities such as `<author>` and `<someone>`. We also do not need to consider normalization step because the direction is given as part of the input. For large language model inference, we use 4-bit quantization (Dettmers et al., 2023) to reduce memory consumption.

Details of computational infrastructures. We use PyTorch (Paszke et al., 2019), Huggingface Transformers (Wolf et al., 2019) and Accelerate (Gugger et al., 2022) to perform model training and inference. All model training is conducted with 1x or 2x Nvidia RTX 3090, or 1x Nvidia RTX A6000. BLOOMZ-176b inference is conducted with 4x Nvidia A100 SMX 40G. The Llama2-70b-chat inference is conducted with 2x Nvidia RTX 3090 or Nvidia A40.

B Experiments on Llama2-70b-chat

Experiment results on Llama2-70b-chat are demonstrated in Table 8. The overall results are similar to the results of BLOOMZ-176b. When requiring directed stance analysis with neutral labels, Llama2-70b-chat with in-context learning provides deficient performance. If we simplify the problem so that output does not require a directed stance, or samples do not include neutral labels, Llama2-70b-chat shows more legitimate performance.

Test Data Type	Micro F_1	Macro F_1
Full Label	18.48	19.86
- w/o Direction	56.56	50.82
- w/o Neutral	30.37	30.63
- w/o Both	84.55	54.92

Table 8: Analysis of Llama2-70b-chat performances on different test data types on DSE. The results show that large language models suffer from predicting the neutral labels and directional information.

BLOOMZ-176b

```
The task is to detect the stance from source entity to target entity given a context. The input consist a pair of entities and a context. Your output can only be "Neutral", "Entity 1 to Entity 2 Positive", "Entity 1 to Entity 2 Negative", "Entity 2 to Entity 1 Positive", "Entity 2 to Entity 1 Negative" without explanation. Below are a few examples:

Context: ...
Entity 1: ...
Entity 2: ...
Stance: ...

Context: ...
...
```

Llama2-70b-chat

```
<s>[INST] <<SYS>>
The task is to detect the stance from source entity to target entity given a context. The input consist a pair of entities and a context. Your output can only be "Neutral", "Entity 1 to Entity 2 Positive", "Entity 1 to Entity 2 Negative", "Entity 2 to Entity 1 Positive", "Entity 2 to Entity 1 Negative" without explanation. Below are a few examples:

Context: ...
Entity 1: ...
Entity 2: ...
Stance: ...

Context: ...
...
<</SYS>>
Context: ...
Entity 1: ...
Entity 2: ...
[/INST] Stance:
```

Figure 4: Prompt templates for BLOOMZ-176b and Llama2-70b-chat.

C Large Language Model Prompts

We demonstrate the prompt templates for large language models in Figure 4. The prompts consist of a short description of the task, with a series of examples. We list the sample to solve at the end of all demonstration examples.

Asked about quarterback (**e1, Colin Kaepernick**) favoriting negative comments on Twitter as a form of personal motivation, (**e2, Harbaugh**) gave it a thumbs up.
Classification Prediction: Neutral
Classification + Consistency Training: $e_2 \xrightarrow{\text{Positive}} e_1$
In the primaries, (**e1, Morell**) said, Putin played upon Mr. (**e2, Trump**)'s vulnerabilities by complimenting him.
Classification Prediction: Neutral
Classification + Consistency Training: $e_1 \xrightarrow{\text{Positive}} e_2$

Table 9: Case study to compare the differences between vanilla classification model and classification model with consistency transitive constrained learning.

D Case Study

We also include two cases from the classification-based method on the DSA dataset to demonstrate the effects of consistency constrained learning, as shown in Table 9. In this two examples, we can find that the consistency learning can help with finding the stance label and direction in the context while the baseline classification model only predicts neutral.