Robust Integration of Contextual Information for Cross-Target Stance Detection

Tilman Beck[*1], Andreas Waldis[*1,2], Iryna Gurevych[1]
[1]Ubiquitous Knowledge Processing Lab (UKP Lab)
Department of Computer Science and Hessian Center for AI (hessian.AI)
Technical University of Darmstadt
[2]Information Systems Research Lab
Department of Computer Science, Lucerne University of Applied Sciences and Arts
www.ukp.tu-darmstadt.de www.hslu.ch

Abstract

Stance detection deals with identifying an author's stance towards a target. Most existing stance detection models are limited because they do not consider relevant contextual information which allows for inferring the stance correctly. Complementary context can be found in knowledge bases but integrating the context into pretrained language models is non-trivial due to the graph structure of standard knowledge bases. To overcome this, we explore an approach to integrate contextual information as text which allows for integrating contextual information from heterogeneous sources, such as structured knowledge sources and by prompting large language models. Our approach can outperform competitive baselines on a large and diverse stance detection benchmark in a cross-target setup, i.e. for targets unseen during training. We demonstrate that it is more robust to noisy context and can regularize for unwanted correlations between labels and target-specific vocabulary. Finally, it is independent of the pretrained language model in use.[1]

1 Introduction

Given a text and a target the text is directed at, stance detection (SD) aims to predict whether the text contains a positive or negative stance towards the target or is unrelated. We provide an example in Figure 1. In contrast to formal polls, stance detection (SD) provides a scalable alternative to assess opinions expressed in unstructured texts. However, in contrast to predicting the polarity of a text (i.e., sentiment analysis), SD is more challenging because it requires establishing the relation towards a target which is rarely mentioned in the text (Augenstein et al., 2016).

Further, to infer the correct stance, often the text alone is not sufficient and contextual information needs to be taken into account (Du Bois, 2007). In contrast, most stance classification models are expected to make a correct prediction given the text and target only. This can lead to overly relying on label correlations with target-specific vocabulary (Reuver et al., 2021; Thorn Jakobsen et al., 2021). In our example (Figure 1), it is challenging to follow the reasoning of the text if the meaning of *school spirit* is left unclear.

Target: School Uniforms
Label: Pro
Text: Creates a sense of school spirit.
Context: ['school spirit is the enthusiasm and pride felt by the students of a school', 'a strong sense of school spirit is a positive and uplifting influence on the school and its students']

Figure 1: Example for Stance Detection from the UKP ArgMin dataset (Stab et al., 2018). The context is not part of the original dataset and was extracted from a large language model via prompting.

Consequently, providing external knowledge as an additional signal to stance classification has been proposed as a remedy. However, lacking a general solution, previous work applies knowledge integration only for a specific text domain like social media (Allaway et al., 2021; Clark et al., 2021). Nevertheless, SD algorithms are applied on a multitude of different text sources like social media (ALD), news (Hanselowski et al., 2019) or debating fora (Hasan and Ng, 2013; Chen et al., 2019) and on diverse targets such as persons (Sobhani et al., 2017; Li et al., 2021), products (Somasundaran and Wiebe, 2010), or controversial topics (Stab et al., 2018; Jo et al., 2021a), among other things. In addition, existing approaches (Zhang et al., 2020; Paul et al., 2020) often depend on the structure of the external knowledge source used. However, a single source of knowledge will likely not suffice for all different scenarios and adapting

[*] Equal contribution.

[1] Data and code at https://github.com/UKPLab/arxiv2022-context-injection-stance

*Proceedings of the The 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 494–511
July 13-14, 2023 ©2023 Association for Computational Linguistics

the model architecture to the structure of a specific knowledge source (e.g. graph-based) limits its applicability.

This work proposes a flexible and robust approach to integrate contextual information by encoding it as text. It is better aligned to the encoding schema of a pre-trained language model (PLM) and circumvents any dependency on the structure of a particular knowledge source. It also allows for using any context source that best fits the data's text domain or mixing contextual information from multiple sources. In detail, we propose a dual-encoder architecture (INJECT), which encodes the input text and context information separately while facilitating information exchange between both via attention. We investigate extracting contextual information from various sources using different extraction strategies. We evaluate our approach across a benchmark of 16 stance detection datasets exhibiting different characteristics concerning text source, size, and label imbalance.

First, we demonstrate that existing state-of-the-art approaches outperform standard baselines only on the domains they have been tuned for - but perform worse on average. When integrating context via INJECT, we observe improvements on average and provide an analysis demonstrating the robustness of our approach. In summary, we make the following contributions:

- We show that the performance of existing state-of-the-art approaches does not transfer across a large and diverse benchmark of 16 SD datasets compared to a standard baseline.
- We propose the INJECT architecture to integrate contextual information for cross-target stance detection. Our approach leads to performance improvements across the benchmark and is independent of the underlying pre-trained language model.
- We compare different sources for extracting contextual information and their effectiveness for stance detection. We extract context from structured knowledge bases by prompting a large pre-trained language model.
- An analysis highlights our approach's benefits compared to a more direct integration via appending the context to the input. Our approach regularizes the influence of correlations of target-specific vocabulary and is robust to noisy contexts.

2 Related Work

Many tasks in NLP benefit from access to external knowledge such as natural language inference (Chen et al., 2018), machine translation (Shi et al., 2016) or argument mining (Lauscher et al., 2022). Within the era of PLMs, many approaches rely on extensive pretraining using data from knowledge bases (Peters et al., 2019; Zhang et al., 2019) (KB) or supervision from knowledge completion tasks (Wang et al., 2021; Rozen et al., 2021).

Early works leveraged sentiment lexicons (Bar-Haim et al., 2017b) or combinations thereof (Zhang et al., 2020) to improve SD classification performance. Other contextual components like author information (Li et al., 2018; Sasaki et al., 2018; Lukasik et al., 2019), dissemination features of social media (Lai et al., 2018; Veyseh et al., 2017) such as retweets or structural discourse elements (Nguyen and Litman, 2016; Opitz and Frank, 2019) have been shown to play an important role for stance detection. Similarly to the aforementioned approaches, the focus in SD has shifted towards combining structural KBs and PLMs. Kawintiranon and Singh (2021) identify label-relevant tokens and prioritize those during masked language modeling. This approach risks overfitting on target-specific tokens because stance is often expressed using target-specific terminology - an issue which is particularly problematic for argumentative sentences (Thorn Jakobsen et al., 2021). Clark et al. (2021) apply a knowledge infusion method for PLMs by filtering Wikipedia triplets for contextual knowledge. Popat et al. (2019) extend BERT by introducing a consistency constraint to learn agreement between the text and its target. Jo et al. (2021b) presents a variant of BERT pre-trained using a variety of supervised tasks resembling logical mechanisms. Paul et al. (2020) extract relevant concepts from ConceptNet using graph-based ranking methods to improve argument relation classification. Likewise, Liu et al. (2021) uses ConceptNet to identify relevant concept-edge pairs and integrate them via a graph neural network during training. Finally, Hardalov et al. (2021) used label embeddings to improve SD multi-dataset learning and recently showed (Hardalov et al., 2022) that sentiment-based pretraining improves multi-lingual stance detection.

In summary, most existing approaches integrate knowledge through extensive pretraining on knowledge-rich data. This does not guarantee improvement of the downstream task they are intended for and requires additional experiments. Another line of work introduces architectural dependencies on the structure of the knowledge source, thereby limiting their usage to tasks and domains for which the knowledge source is applicable. In contrast, our approach does not require further pretraining but directly learns to integrate contextual information during supervised training. The usefulness of the context is, therefore, directly measurable. Further, our proposed approach integrates context in natural language, thereby decoupling it from the structure of the context source. It is better aligned with the encoding mechanism of pre-trained language models and enables the integration of contextual information from various sources.

3 Methodology

Our goal is twofold: (1) we aim to integrate contextual information independent of the context source and (2) in a way that is robust to noisy and irrelevant content in the context. We propose INJECT, a dual encoder approach to integrate contextual sentences using the cross-attention mechanism introduced by Vaswani et al. (2017). The general intuition is that information can flow from input to context and vice versa, thereby regularizing the attention in both encoders. Thus, the context provides further information to reweigh the prediction importance of individual tokens in the input.

3.1 Contextual information

With regards to stance detection, we define *contextual information* (or short *context*) as the sum of all information which, given the text and its target, renders the conclusion of the stance plausible. The context for each dataset instance is retrieved beforehand and is provided as text to the model. Formally, we describe context $\mathbf{c} \in \mathbf{C}$ where $\mathbf{c}$ is a list containing m texts which provide contextual information on the input text $\mathbf{x}$. See Figure 1 for an example ($m = 2$). The length of these texts is upper bounded by the maximum sequence length of the encoder model.

3.2 Context integration via INJECT

Figure 2 provides a high-level visualization of our proposed INJECT architecture. It consists of two modules: input- and context-encoder. The input encoder processes input and target $I = (X, T)$ while the context encoder processes the context sentences C. The encoders exchange information using inject blocks ($IB^{(j)}$) which are injected on layer j of both encoders. j is a hyperparameter tuned using the dataset's development set. All other layers are standard transformer blocks. An IB block is technically similar to a self-attention block but receives different inputs for key K, value V, and query Q. In detail, the inject block of the context-encoder (on layer j) receives the output from the self-attention $e^{(j)}_{(I,s)}$ on layer j of the input-encoder as key and value and the output of its own self-attention $e^{(j)}_{(C,s)}$ on layer j as query:

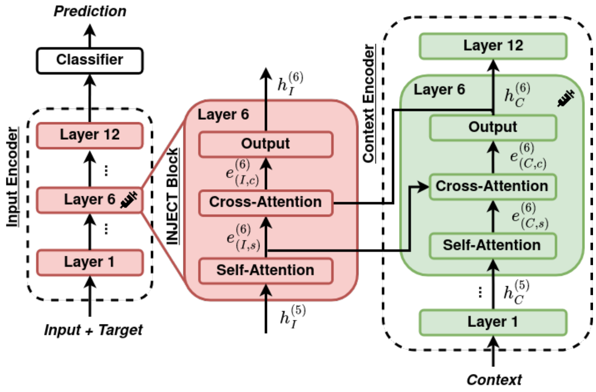

Figure 2: Visualization of the INJECT architecture. It consists of two modules - input encoder and context encoder. The context encoder encodes contextual information, and both encoders are interwoven using an INJECT-block based on the cross-attention mechanism.

$$IB^{(j)}(K{=}e^{(j)}_{(I,s)}, V{=}e^{(j)}_{(I,s)}, Q{=}e^{(j)}_{(C,s)})$$

Afterward, it is forwarded to get the new hidden state $h_C^{(i)}$ of the context. Next, we back-inject the context into the input-encoder by feeding $h_C^{(i)}$ as key and value in its inject block:

$$IB^{(j)}(K{=}h_C^{(j)}, V{=}h_C^{(j)}, Q{=}e^{(j)}_{(I,s)})$$

Next, the hidden state $h_I^{(i)}$ at layer j of the input encoder is produced by processing the cross-attention output $e^{(j)}_{(I,c)}$. Finally, we add a classification head to the input encoder, which consists of a pooling layer, a dropout, and a linear classification layer. The parameters of both modules are optimized using the standard cross-entropy loss.

Our architecture is flexible regarding the number of context sentences that can be encoded (parameter m). In the case of multiple sentences, we average the cross-attention for all of them. Due to the dual encoders, INJECT is computationally more efficient than context integration via concatenation, as we explain in the Appendix A.7.

3.3 Context integration via concatenation

An alternative approach would be to append contextual information to the input text such that the model can exploit context directly using the self-attention mechanism. Technically, this is implemented by separating the input and context using the model-specific separation token (e.g., `text` + `[SEP]` + `context` for BERT)[2].

We see two major drawbacks of this approach. First, integrating irrelevant context will hurt downstream performance due to its direct influence on attention. Second, it is limited by the maximum sequence length of the model in use.

4 Context integration for stance detection

Task In stance detection, given an input text $\mathbf{x} \in \mathbf{X}$ and its corresponding target $\mathbf{t} \in \mathbf{T}$, the goal is to identify the correct label $\mathbf{y} \in \mathbf{Y}$ from a predefined set of stance descriptions. We further provide a set of contextual sentences C. The retrieval of C is explained in the next section.

4.1 Context retrieval

The INJECT model expects the context in natural language form and is therefore flexible with regard to the source of contextual information. To showcase, we evaluate different context sources that we deem relevant for inferring stance relations: (1) a structured knowledge base which stores knowledge as entity-relationship triplets, (2) a set of causal relations extracted from an encyclopedia, and (3) prompting a large pretrained language model using predefined question templates. The latter provides an intuitive interface to prompt for relevant sample-specific context, especially without suitable knowledge bases.

We neither expect these sources always to provide *perfect* contextual information nor to be suitable for all of the heterogeneous stance detection applications (see §5.1). However, our proposed architecture is designed to be robust, i.e., it utilizes beneficial context and ignores irrelevant information. In the following, we describe each approach in detail.

ConceptNet Oftentimes, commonsense knowledge is beneficial to infer the stance towards a target correctly and has been shown to complement stance classification (Liu et al., 2021). Therefore, we use ConceptNet (Speer et al., 2017), a directed graph whose nodes are concepts and whose edges are assertions of commonsense about these concepts. For every edge, ConceptNet provides a textual description of the type of node relationship. Further, ConceptNet provides a weight factor for every edge computed based on the edge frequency within the ConceptNet training corpus.

Our approach uses the English subset of ConceptNet to get context sentences. We filter out concepts that are part of English stopwords [3] and ignore relations without descriptions. In total, we consider 400k nodes connected through approximately 600k edges. To retrieve the context, we use all tokens of the input text to search for string matches within the ConceptNet concepts. We consider only paths of length one where the start- and/or end-concept are contained in the input text. Finally, we sort the paths based on their weight (provided by ConceptNet) and convert every path into a context candidate by joining the descriptions of all its edges, as done in previous work (Lauscher et al., 2020).

CauseNet Causal relations, as a more specific example of commonsense knowledge, are often beneficial for understanding opinionated expressions (Sasaki et al., 2016) but rarely formulated in the text. To explain such relations, we investigate CauseNet (Heindorf et al., 2020), a KB of claimed causal relations extracted from the ClueWeb12 corpus and Wikipedia. We use the causal relations contained in the high-precision subset[4] of CauseNet, consisting of 80,223 concepts and 199,806 relations. We ignore concepts shorter than three characters or consisting of a modal verb (see Appendix A.6.1). We encode all relations using a sentence encoder (Reimers and Gurevych, 2019) using BERT-base-uncased weights. For each sample in a dataset, we retrieve the most relevant relations by ranking based on the cosine similarity between the encoded sample and all relations.

Pretrained language model Large PLMs can be queried as KBs using natural language prompts (Petroni et al., 2019; Heinzerling and Inui,

[2] In case of two input texts, the context is concatenated to the second input text.

[3] As in NLTK (Bird, 2006)

[4] see `https://causenet.org/`

2021). We adopt this paradigm and generate context candidates by prompting a PLM to provide more information on either the target, parts of the input, or a combination of both. Precisely, we extract noun-phrases from the input sentence of a length of up to three words using the Stanford CoreNLP tool (Manning et al., 2014), ignoring stopwords and filtering noun-phrases that are equal to the target. Then, we create prompts using the following templates for single inputs a (e.g., target or noun-phrase)

$$P_1(a) = \text{define } a$$
$$P_2(a) = \text{what is the definition of } a$$
$$P_3(a) = \text{explain } a$$

and combination of inputs (a, b).

$$P_4(a, b) = \text{relation between } a \text{ and } b$$
$$P_5(a, b) = \text{how is } a \text{ related to } b$$
$$P_6(a, b) = \text{explain } a \text{ in terms of } b$$

The single-input approach is referred to as `T0pp-NP`, and the second approach as `T0pp-NP-T`. We found those prompts to generate the most meaningful contexts across different targets and noun-phrases (see Appendix A.6.2 for more details). The prompts can then generate outputs using any pre-trained sequence-to-sequence model.

We make use of T0pp[5] (Sanh et al., 2022), which is based on a pre-trained encoder-decoder (Raffel et al., 2020) and was fine-tuned using multiple diverse prompts generated using a large set of supervised datasets. We set the output sequence length to 40 words and sort the generated outputs by the length in descending order because we sometimes observe T0pp degenerate into producing single words. We filter those candidates where more than half of the generated words are repetitions. Finally, we remove all special tokens from the candidates. We found using two context sentences ($m = 2$) most beneficial in preliminary experiments.

5 Experiments

5.1 Datasets

We use a SD benchmark (Schiller et al., 2021; Hardalov et al., 2021), which covers 16 English datasets for research on (cross-domain) stance detection. We use this benchmark (Table 1) because it shows a large diversity regarding text sources, the number of targets, the number of annotated instances, and label imbalance. Thus, it provides a suitable testbed to evaluate the effectiveness of our context injection approach. More information about the details of each dataset can be found in the Appendix A.2.

5.2 Experimental details

Evaluation Our results are evaluated in a cross-target fashion (Augenstein et al., 2016; Xu et al., 2018), i.e., the setup is organized such that instances of a specific target are only contained in the training, development, or test split. We point out that our results are not directly comparable to Hardalov et al. (2021) as they perform experiments in a cross-domain fashion, i.e., their goal is to evaluate transfer learning effects by training on *all* but one dataset, which is used for testing. In contrast, we use *one* dataset per experiment to study the usefulness of context integration.

Baselines We compare INJECT to the following baselines. `BERT` is provided only the input, whereas `BERT+Target` is provided with both input and target using the model-specific separator token. (`BERT`$\oplus$`X`) refers to `BERT+Target` with the retrieved context being appended, where X refers to context sources used (`ConceptNet`, `CauseNet`, `T0pp-NP` and `T0pp-NP-T`). We also test a combination of all context sources (`All`) and integration of random context (`Random`). To the best of our knowledge, no prior work has evaluated context integration for cross-target SD on the full benchmark. Thus, we compare with `TGA-Net` (Allaway and McKeown, 2020), `STANCY` (Popat et al., 2019), and `JointCL` (Liang et al., 2022) three state-of-the-art methods for SD which have been proposed for subsets of the benchmark. `TGA-Net` uses clustering to identify generalized topic representations. `STANCY` applies contrastive learning to learn embeddings where texts supporting a target are closer and opposing texts are more distant to their targets. `JointCL` use a prototypical graph for target-aware token representations. All of them require target information, which is not available for semeval19. In addition, we found `JointCL` is not working on fnc1 due to its long input texts. In these cases, we use the corresponding `BERT-Target` score for macro-F_1 avg. calcula-

[5] See `https://huggingface.co/bigscience/T0pp`. We also did experiments using T5 directly but found the outputs to be inferior to T0pp.

Dataset	**Target**	**Type**	**Labels**	**Source**
arc (Habernal et al., 2018)	Headline	User Post	unrelated (75%), disagree (10%), agree (9%), discuss (6%)	Debates
iac1 (Walker et al., 2012)	Topic	Debating Thread	pro (55%), anti (35%), other (10%)	Debates
perspectrum (Chen et al., 2019)	Claim	Perspective Sent.	support (52%), undermine (48%)	Debates
poldeb (Somasundaran and Wiebe, 2010)	Topic	Debate Post	for (56%), against (44%)	Debates
scd (Hasan and Ng, 2013)	None (Topic)	Debate Post	for (60%), against (40%)	Debates
emergent (Ferreira and Vlachos, 2016)	Headline	Article	for (48%), observing (37%), against (15%)	News
fnc1 (Pomerleau and Rao, 2017)	Headline	Article	unrelated (73%), discuss (18%), agree (7%), disagree (2%)	News
snopes (Hanselowski et al., 2019)	Claim	Article	agree (74%), refute (26%)	News
mtsd (Sobhani et al., 2017)	Person	Tweet	against (42%), favor (35%), none (23%)	Social Media
rumor (Qazvinian et al., 2011)	Topic	Tweet	endorse (35%), deny (32%), unrelated (18%), question (11%), neutral (4%)	Social Media
semeval2016t6 (Mohammad et al., 2016)	Topic	Tweet	against (51%), none (24%), favor (25%)	Social Media
semeval2019t7 (Gorrell et al., 2019)	None (Topic)	Tweet	comment (72%), support (14%), query (7%), deny (7%)	Social Media
wtwt (Conforti et al., 2020)	Claim	Tweet	comment (41%), unrelated (38%), support (13%), refute (8%)	Social Media
argmin (Stab et al., 2018)	Topic	Sentence	argument against (56%), argument for (44%)	Various
ibmcs (Bar-Haim et al., 2017a)	Topic	Claim	pro (55%), con (45%)	Various
vast (Allaway and McKeown, 2020)	Topic	User Post	con (39%), pro (37%), neutral (23%)	Various

Table 1: Stance Detection Benchmark datasets and their characteristics (sorted by source, then alphabetically). This table is based on Hardalov et al. (2021).

	macro-F$_1$avg.	arc	iac1	perspectrum	poldeb	scd	emergent	fnc1	snopes	mtsd	rumor	semeval16	semeval19	wtwt	argmin	ibmcs	vast
`BERT`	48.3±0.7	21.5	35.6	64.6	**51.3**	56.7	78.3	27.2	68.7	40.4	44.6	63.5	53.7	25.5	59.6	50.7	32.2
`BERT+Target`	56.8±0.8	62.5	36.3	**76.0**	49.8	57.9	78.0	72.9	69.7	41.2	40.5	64.8	53.7	55.2	60.3	52.0	36.1
`STANCY`	56.2±0.5	62.6	36.9	75.2	50.2	57.9	78.3	**74.3**†	**69.9**	40.3	32.9	64.9	–	54.0	60.0	52.5	36.1
`TGA-Net`	46.8±1.4	57.2	33.9	57.5	42	49.8	59.0	46.2	57.1	37.7	16.0	59.5	–	19.0	50.1	47.9	**62.7**†
`JointCL`	50.9±1.8	28.6	35.8	69.6	27.2	47.1	**78.9**	–	69.7	**55.1**†	51.5†	**67.5**†	–	**65.1**†	35.3	35.3	31.4
`BERT⊕ConceptNet`	55.7±0.6	61.4	**39.3**†	74.2	49.2	57.6	76.4	72.1	69.4	41.1	44.6	63.5	53.3	43.5	60.2	50.0	35.1
`BERT⊕CauseNet`	54.9±1.3	60.6	35.0	74.4	50.0	58.0	75.0	70.9	69.2	43.2	39.1	61.1	54.3	44.5	59.4	47.3	36.0
`BERT⊕T0pp-NP`	55.7±1.0	61.3	37.2	74.0	49.8	54.5	77.2	71.9	69.4	42.1	41.3	62.4	52.2	50.9	60.2	51.1	35.4
`BERT⊕T0pp-NP-T`	56.2±0.8	61.4	36.7	73.3	48.8	58.2	77.5	72.1	69.8	40.6	44.5	61.9	53.5	54.2	59.3	52.2	34.4
`BERT⊕All`	55.5±1.3	61.5	38.2	74.3	49.5	56.2	75.7	70.9	68.8	43.5	42.7	62.4	**55.3**†	42.9	60.3	50.6	35.5
`BERT⊕Random`	54.5±1.1	61.3	36.3	74.5	49.7	48.3	74.8	72.1	69.6	38.4	38.2	61.8	53.8	49.6	59.0	48.2	36.1
`BERT⊗ConceptNet`	57.2±1.0	62.7	36.5	75.6	49.3	**58.3**	77.8	73.8	69.0	41.9	47.9	65.1	54.4	52.5	60.1	53.0	37.4†
`BERT⊗CauseNet`	57.7±0.9	62.9	36.9	75.5	48.9	58.0	78.1	73.6	69.3	42.4	48.1	65.7†	55.1	54.8	60.7	**53.6**†	39.6†
`BERT⊗T0pp-NP`	57.5±1.0	62.6	37.2	75.6	48.7	57.2	77.2	73.7	69.6	41.2	49.2†	65.6†	55.1	55.6	60.9	52.9	37.3†
`BERT⊗T0pp-NP-T`	**57.8±1.0**	62.7	37.2	75.9	49.1	57.9	78.7	74.0†	69.1	41.4	**52.2**†	65.9†	55.0	54.4	**61.0**	53.4†	37.5†
`BERT⊗All`	57.2±0.9	**63.0**	36.6	75.4	49.7	57.9	78.8	73.3	69.1	42.4	44.3	65.5	54.8	53.6	60.3	53.6†	37.7†
`BERT⊗Random`	57.3±1.0	62.9	36.8	75.5	49.4	57.9	78.0	73.5	69.6	41.6	45.4	65.8	54.3	54.4	60.5	53.4†	37.5†

Table 2: Overview of the cross-target results across stance detection benchmark datasets. We highlight best performance per evaluation setting and dataset in bold. Statistically significant differences compared to the best performing baseline without access to context (`BERT+Target`) are indicated by †. Numbers are macro-F$_1$ scores averaged over ten runs with differently initialized seeds.

tion for a fair comparison.

Training setup We make use of the standard splits given in the benchmark (Hardalov et al., 2021) where possible or create our own (see Appendix A.1). We use macro-F$_1$ as evaluation metric and average across ten runs with different seeds. Performance is measured after the best-performing epoch based on the development set. We use Mann-Whitney U-Test (Mann and Whitney, 1947) with $p < 0.05$ to test statistical significance. We use the uncased BERT base model (Devlin et al., 2019) for all experiments. We use the same set of hyper-parameters for all model setups. For INJECT, we use the same model architecture for the input and context encoder. We tune the layer j for context integration (see §3.2). We tested layers 3, 6, 9, and 12 on the development set of the benchmark. Layer 12 performed the best and was used for all reported results. More details are in the Appendix A.1.

6 Results

This section shows the effectiveness of INJECT by providing constant improvements using noisy context on the heterogeneous benchmark (Table 2).

First, we note a large performance boost (+8.5pp) when including information about the target when comparing `BERT` and `BERT+Target`. While target information improved the performance for individual datasetes (Stab et al., 2018), we generalize this finding for 14 out of 16 SD datasets.

The baselines `STANCY`, `TGA-Net`, and `JointCL` mostly show the best performance for the datasets they have been proposed for. However, on average, they do not perform on par with the strong `BERT+Target` baseline. `STANCY` performs slightly worse, probably due to the binary contrastive loss and thereby ignoring multi-label information. TGA-Net underperforms all other approaches except for vast. Using generalized topic representations transfers to a scenario where the number of targets is relatively high (5634) and only a few examples per target exist (mean 2.4), as for vast. JointCL performs best on datasets from social media (semeval16, wtwt, or mtsd), but is outperformed by standard baselines for the rest of the tasks. Thus, this approach can not generalize to datasets with longer text inputs. We conclude that existing state-of-the-art approaches for cross-target stance detection work well for the datasets they have been designed for but do not generalize across the diverse set of datasets that exist in SD.

INJECT outperforms BERT+Target in 13 of 16 cases, while for three datasets (perspectrum, poldeb, snopes) none of the extracted contexts provides benefits, independently of the integration. On average, all context sources lead to performance improvements in combination with INJECT, with T0pp-NP-T being the best. Combining all context sources underperforms the integration of individual context, most probably due to the average function leading to a perturbation of the context. Surprisingly, integration of random context slightly outperforms the strong BERT+Target baseline in ten datasets while degrading the direct integration performance, as expected. We investigate the reasons in our subsequent analysis (§6.3).

6.1 Quality of context

To evaluate the quality of each context source, we looked at the aggregated performance differences with the baseline across each source (Figure 3b). While CauseNet leads to performance improvements for a maximum number of tasks (12), T0pp-NP-T leads the board concerning the total sum of absolute improvements across all datasets. The context quality extracted by prompting a PLM also becomes evident when looking at the performance of BERT⊕. ConceptNet and CauseNet lead to large performance degradation both in number of tasks and absolute numbers.

6.2 Generalization across PLMs

We investigate if the benefits of INJECT transfer to other PLM architectures by evaluating it in combination with RoBERTa (Liu et al., 2019) and ELECTRA (Clark et al., 2020). We follow the same experimental protocol as for BERT A.1, but chose only the best-performing context source (T0pp-NP-T) due to the large number of experiments. The results (Table 3) confirm the previously observed findings that INJECT improves the performance on average across this diverse set of stance detection tasks. We observe similar improvements as with BERT for both models, with the strongest increase (+1.1pp on average) and the best overall performance for ELECTRA.

6.3 Further analysis

As integration with INJECT outperforms direct integration and even performs more robustly when provided with random context, we analyze the regularization provided by the INJECT architecture.

Regularization via INJECT We analyze how INJECT regularizes inputs by examining how models rely on target-specific vocabulary. Such vocabulary is often used to express a stance (Wei and Mao, 2019), but can lead to spurious correlations (Thorn Jakobsen et al., 2021). Therefore, we identify the top 5% label-indicative and target-specific tokens and correlate them with the model attributions using vector-norms (Kobayashi et al., 2020) (see Appendix A.4 for details). Table 4 shows these correlations for six benchmark datasets. For arc, argmin, and rumor we note a general low to negative correlation of BERT+Target. Further, we see BERT⊕T0pp-NP-T and BERT⊗T0pp-NP-T increasing the correlation - giving more attribution to target and label indicative tokens. This behavior is one reason for the bad performance of BERT+Target for these tasks. For rumor, we note a correlation increase of 45.5 for INJECT+T0pp-NP-T, which leads to a clear performance improvement of 11.7pp (Table 2). Thus, INJECT adjusts the low attribution to relevant tokens compared to BERT+Target. On the other hand, we see INJECT+T0pp-NP-T reducing the attribution for ibmcs, mtsd, and wtwt. Given the better performance of INJECT+T0pp-NP-T on ibmcs and mtsd, we conclude that INJECT can reduce potential spurious correlations in this case. For wtwt, INJECT+T0pp-NP-T reduces the correlation but has a performance loss of 0.8pp. Given the niche domain of wtwt (financial mergers and acquisitions on Twitter), identifying relevant context is more challenging using standard context sources.

Dataset characteristics We investigate which dataset characteristics are indicative of perfor-

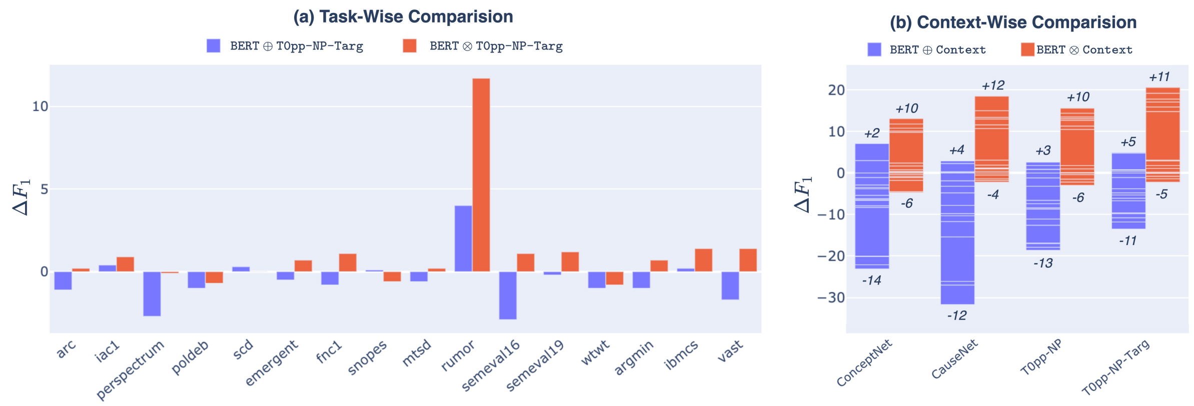

Figure 3: In (a), we compare the relative performance change ΔF_1 of `BERT⊕T0pp-NP-T` (blue) or `BERT⊗T0pp-NP-T` (red) compared to `BERT+Target` for every task. Within (b), we show the aggregated relative performance change of `BERT⊕`(blue) and `BERT⊗`(red) compared to `BERT+Target` per context source. In addition, we count the number of tasks exhibiting performance improvement and deterioration above and below the bars, respectively.

	macro-F_1avg.	arc	iac1	perspectrum	poldeb	scd	emergent	fnc1	snopes	mtsd	rumor	semeval16	semeval19	wtwt	argmin	ibmcs	vast
`BERT+Target`	56.8±0.8	62.5	36.3	**76.0**	**49.8**	57.9	78.0	72.9	69.7	41.2	40.5	64.8	53.7	**55.2**	60.3	52.0	36.1
`BERT⊕T0pp-NP-T`	56.2±0.8	61.4	36.7	73.3	48.8	**58.2**	77.5	72.1	**69.8**	40.6	44.5	61.9	53.5	54.2	59.3	52.2	34.4
`BERT⊗T0pp-NP-T`	**57.8±1.0**	**62.7**	**37.2**	75.9	49.1	57.9	**78.7**	**74.0**†	69.1	**41.4**	**52.2**†	**65.9**†	**55.0**	54.4	**61.0**	**53.4**†	**37.5**†
`RoBERTa+Target`	61.6±0.6	60.4	32.9	85.1	49.6	**62.3**	**79.0**	**77.3**	**74.9**	61.2	49.9	70.3	57.8	**64.2**	60.9	62.9	37.0
`RoBERTa⊕T0pp-NP-T`	60.8±0.8	61.7	**35.1**	84.1	**50.6**	62.1	77.8	77.0	73.9	55.2	**51.3**	68.2	57.8	63.4	**61.6**	57.9	35.6
`RoBERTa⊗T0pp-NP-T`	**61.9±0.7**	**62.9**†	33.4	**85.4**	49.6	59.5	78.5	**77.3**	74.6	**64.4**	51.2	**70.5**	**58.0**	62.2	61.1	**63.5**	**38.5**†
`ELECTRA+Target`	62.0±0.9	59.5	35.2	89.2	45.7	61.7	77.4	73.8	75.4	66.9	50.0	**70.1**	55.0	63.7	60.2	71.6	36.2
`ELECTRA⊕T0pp-NP-T`	61.6±0.7	59.6	**35.5**	87.5	**47.8**	**62.1**	77.4	73.8	74.0	64.7	53.1	67.2	54.1	**65.3**	60.6	68.4	35.2
`ELECTRA⊗T0pp-NP-T`	**63.1±0.6**	**62.5**†	35.4	**89.3**	47.4	60.4	**78.2**	**76.2**†	**75.7**	**68.9**†	**54.7**	70.0	**57.1**	63.7	**60.7**	**71.7**	**37.2**

Table 3: Comparing context integration using different PLM architectures in a cross-target setup across stance detection benchmark datasets. We highlight the best performance per model architecture and dataset in bold. Statistically significant differences compared to the best-performing baseline without access to context (`BERT+Target`) are indicated by †. Numbers are macro-F_1 scores averaged over three runs with differently initialized seeds (see Appendix A.1 for experimental details.)

model	arc	rumor	argmin	ibmcs	mtsd	wtwt
`BERT+Target`	-6.3	-14.1	6.9	27.0	64.5	11.6
`BERT⊕T0pp-NP-T`	4.5	31.0	16.2	25.7	48.9	5.3
`BERT⊗T0pp-NP-T`	8.8	31.4	9.8	22.7	33.8	6.4

Table 4: Pearson correlation between self-attention and target-label-specific tokens for the baseline model `BERT+Target` and the context integration approaches (`BERT⊕` and `BERT⊗`) using the best performing context source (`T0pp-NP-T`). A larger correlation indicates stronger attention attribution.

mance improvements using INJECT. Thus, we compute the Pearson correlation of various dataset characteristics and the performance differences between the baseline and the average of the INJECT variants. Details about how we calculate dataset characteristics are provided in the Appendix A.5. The results are visualized in Figure 4. Independent of the context source, we observe beneficial improvements using INJECT if datasets exhibit characteristics leading to performance instabilities. This is indicated by positive correlations with an increasing number of labels and label imbalance. Further, we measure text understanding difficulty using the Flesch reading-ease score (FRES) by Flesch (1948). Interestingly, INJECT can better deal with datasets exposing a high variability of FRES within their instances (mean-flesch, std-flesch). These factors generally contribute to training instabilities where INJECT is more robust. This observation is confirmed by the strong positive correlation of the variance across random initializations and INJECT performance.

Robustness We investigate robustness across all benchmark datasets for the `T0pp-NP-T` context by visualizing the performance differences to the baseline (`BERT+Target`) in Figure 3a. In the case of performance improvements, INJECT consistently outperforms direct context integration. If there is no improvement for both integration ap-

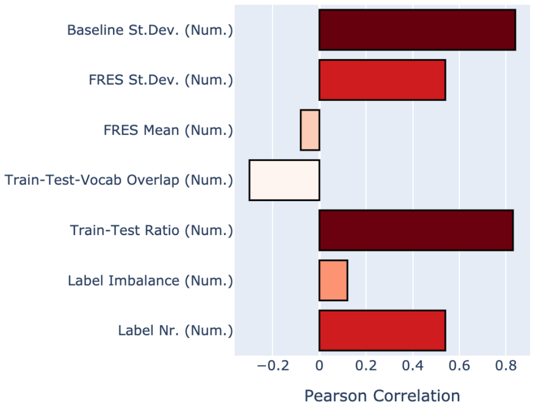

Figure 4: Pearson correlation of various dataset characteristics with performance difference compared to the baseline.

proaches, the performance loss is less pronounced for INJECT with only one exception (snopes). To substantiate this finding, we contrast both context integration approaches in a scenario with both *ideal* context, i.e., the contextual information is guaranteed to be beneficial in predicting the correct class and random context. Our results demonstrate INJECT successfully leveraging the contextual information while not outperforming direct integration in the case of ideal context. However, when given irrelevant context, INJECT is closer to context-free baseline performance. Details about the experiments are provided in the Appendix §A.3. In summary, we conclude that context integration is more robust regarding noisy context.

7 Conclusion

We propose INJECT, a dual-encoder approach to integrate contextual information for stance detection based on cross-attention. While state-of-the-art approaches perform mostly well on the datasets they have been proposed for, we evaluate our approach across a large and diverse benchmark in a cross-target setting and observe improvements compared using three different sources for extracting contextual information. We show that the context integrated via INJECT improves stance detection and is beneficial for generalization on targets not seen during training. In future, we plan to explore more sophisticated ways of prompting large pre-trained language models for helpful context.

Ethical Considerations and Limitations

Quality of the context The performance improvement for contextual information injection is bounded by the quality of the context source. Independently of the source in use, it is possible to introduce additional noise into the training procedure. While this is a rather generic problem, our proposed architecture seems to be better at filtering noisy context than a direct integration via appending to the input.

Quality of context source Most of the existing knowledge bases provide high-quality and curated knowledge. In contrast, when prompting a large language model for knowledge, we are also exposed to the risk that we extract the biases (e.g. false facts or stereotypical biases) that the model has learned during pretraining. In our experiments, we use the T0pp language model where biases have been reported to exist[6]. These biases can potentially influence the prediction performance unintendedly, especially as in many SD datasets, the annotated targets are often controversial. While investigating such effects is out of scope for this work, we consider such an evaluation inevitable before deploying our proposed model to any data outside (academic) research context.

Limitations As described in §3, our proposed approach uses two parallel encoder models (input and context). It thus requires twice as many parameters as the baseline model we compare to, thereby enforcing additional hardware demands. We consider our approach as a proof-of-concept on how to integrate contextual knowledge without amplifying a model's exploitation of spurious correlations. We plan to make our architecture more parameter-efficient by investigating more recent approaches for parameter sharing, e.g. with the use of adapters (Houlsby et al., 2019).

Moreover, we acknowledge the strong influence of wording in prompts on the output of a language model, as has been reported in the literature (Jiang et al., 2020; Schick and Schütze, 2021). We experienced similar effects during preliminary experiments and pointed out that we did not find a one-size-fits-all solution which works equally well across the diverse set of SD benchmark datasets. Therefore, special care must be taken when extracting contextual information from large language models using prompting.

[6] More details at `https://huggingface.co/bigscience/T0pp?`

Acknowledgements

We thank Leonardo F. R. Ribeiro, Haritz Puerto, Nico Daheim and the anonymous ARR reviewers for their valuable feedback. This work has been funded by the German Research Foundation (DFG) as part of the Research Training Group KRITIS No. GRK 2222 and by the German Federal Ministry of Education and Research (BMBF) under the promotional reference 01UP2229B (KoPoCoV) as well as by the Hasler Foundation Grant No. 21024.

References

Stance detection on social media: State of the art and trends. *Information Processing & Management*, 58(4).

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.

Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. 2017a. Stance classification of context-dependent claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 251–261, Valencia, Spain. Association for Computational Linguistics.

Roy Bar-Haim, Lilach Edelstein, Charles Jochim, and Noam Slonim. 2017b. Improving claim stance classification with lexical knowledge expansion and context utilization. In *Proceedings of the 4th Workshop on Argument Mining*, pages 32–38, Copenhagen, Denmark. Association for Computational Linguistics.

Vasant P. Bhapkar. 1966. A note on the equivalence of two test criteria for hypotheses in categorical data. *Journal of the American Statistical Association*, 61:228–235.

Steven Bird. 2006. NLTK: The Natural Language Toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72, Sydney, Australia. Association for Computational Linguistics.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9539–9549. Curran Associates, Inc.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2406–2417, Melbourne, Australia. Association for Computational Linguistics.

Sihao Chen, Daniel Khashabi, Wenpeng Yin, Chris Callison-Burch, and Dan Roth. 2019. Seeing things from a different angle:discovering diverse perspectives about claims. 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 542–557, Minneapolis, Minnesota. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pretraining text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Thomas Clark, Costanza Conforti, Fangyu Liu, Zaiqiao Meng, Ehsan Shareghi, and Nigel Collier. 2021. Integrating transformers and knowledge graphs for Twitter stance detection. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 304–312, Online. Association for Computational Linguistics.

Costanza Conforti, Jakob Berndt, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd, and Nigel Collier. 2020. Will-they-won't-they: A very large dataset for stance detection on Twitter. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1724, Online. Association for Computational Linguistics.

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.

John W Du Bois. 2007. The stance triangle. *Stancetaking in discourse: Subjectivity, evaluation, interaction*, 164(3):139–182.

William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1168, San Diego, California. Association for Computational Linguistics.

Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. 2019. SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. 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 1930–1940, New Orleans, Louisiana. Association for Computational Linguistics.

Andreas Hanselowski, Christian Stab, Claudia Schulz, Zile Li, and Iryna Gurevych. 2019. A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 493–503, Hong Kong, China. Association for Computational Linguistics.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2021. Cross-domain label-adaptive stance detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9011–9028, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2022. Few-shot cross-lingual stance detection with sentiment-based pre-training. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 10729–10737. AAAI Press.

Kazi Saidul Hasan and Vincent Ng. 2013. Stance classification of ideological debates: Data, models, features, and constraints. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1348–1356, Nagoya, Japan. Asian Federation of Natural Language Processing.

Stefan Heindorf, Yan Scholten, Henning Wachsmuth, Axel-Cyrille Ngonga Ngomo, and Martin Potthast. 2020. Causenet: Towards a causality graph extracted from the web. In *Proceedings of the 29th ACM International Conference on Information I& Knowledge Management*, CIKM'20, page 3023–3030, New York, NY, USA. Association for Computing Machinery.

Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1772–1791, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How Can We Know What Language Models Know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Yohan Jo, Seojin Bang, Chris Reed, and Eduard Hovy. 2021a. Classifying Argumentative Relations Using Logical Mechanisms and Argumentation Schemes. *Transactions of the Association for Computational Linguistics*, 9:721–739.

Yohan Jo, Seojin Bang, Chris Reed, and Eduard Hovy. 2021b. Classifying argumentative relations using logical mechanisms and argumentation schemes. *Transactions of the Association for Computational Linguistics*, 9:721–739.

Kornraphop Kawintiranon and Lisa Singh. 2021. Knowledge enhanced masked language model for stance detection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4725–4735, Online. Association for Computational Linguistics.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, Online. Association for Computational Linguistics.

Mirko Lai, Viviana Patti, Giancarlo Ruffo, and Paolo Rosso. 2018. Stance evolution and twitter interactions in an italian political debate. In *Natural Language Processing and Information Systems*, pages 15–27, Cham. Springer International Publishing.

Anne Lauscher, Olga Majewska, Leonardo F. R. Ribeiro, Iryna Gurevych, Nikolai Rozanov, and Goran Glavaš. 2020. Common sense or world knowledge? investigating adapter-based knowledge injection into pretrained transformers. In *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 43–49, Online. Association for Computational Linguistics.

Anne Lauscher, Henning Wachsmuth, Iryna Gurevych, and Goran Glavaš. 2022. Scientia potentia est—on the role of knowledge in computational argumentation. *Transactions of the Association for Computational Linguistics*, 10:1392–1422.

Chang Li, Aldo Porco, and Dan Goldwasser. 2018. Structured representation learning for online debate stance prediction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3728–3739, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yingjie Li, Tiberiu Sosea, Aditya Sawant, Ajith Jayaraman Nair, Diana Inkpen, and Cornelia Caragea. 2021. P-stance: A large dataset for stance detection in political domain. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2355–2365, Online. Association for Computational Linguistics.

Bin Liang, Qinglin Zhu, Xiang Li, Min Yang, Lin Gui, Yulan He, and Ruifeng Xu. 2022. 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.

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.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Michal Lukasik, Kalina Bontcheva, Trevor Cohn, Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2019. Gaussian processes for rumour stance classification in social media. *ACM Trans. Inf. Syst.*, 37(2):20:1–20:24.

Henry B Mann and Donald R Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. 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.

Huy Nguyen and Diane Litman. 2016. Context-aware argumentative relation mining. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1127–1137, Berlin, Germany. Association for Computational Linguistics.

Juri Opitz and Anette Frank. 2019. Dissecting content and context in argumentative relation analysis. In *Proceedings of the 6th Workshop on Argument Mining*, pages 25–34, Florence, Italy. Association for Computational Linguistics.

Debjit Paul, Juri Opitz, Maria Becker, Jonathan Kobbe, Graeme Hirst, and Anette Frank. 2020. Argumentative relation classification with background knowledge. In *Computational Models of Argument - Proceedings of COMMA 2020, Perugia, Italy, September 4-11, 2020*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 319–330. IOS Press.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word 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 43–54, Hong Kong, China. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and

Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Dean Pomerleau and Delip Rao. 2017. Fake news challenge stage 1 (FNC-I): Stance detection.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2019. STANCY: Stance classification based on consistency cues. 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 6413–6418, Hong Kong, China. Association for Computational Linguistics.

Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. 2011. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Myrthe Reuver, Suzan Verberne, Roser Morante, and Antske Fokkens. 2021. Is stance detection topic-independent and cross-topic generalizable? - a reproduction study. In *Proceedings of the 8th Workshop on Argument Mining*, pages 46–56, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ohad Rozen, Shmuel Amar, Vered Shwartz, and Ido Dagan. 2021. Teach the rules, provide the facts: Targeted relational-knowledge enhancement for textual inference. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 89–98, Online. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen 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 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 Fevry, 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 *International Conference on Learning Representations*.

Akira Sasaki, Kazuaki Hanawa, Naoaki Okazaki, and Kentaro Inui. 2018. Predicting stances from social media posts using factorization machines. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3381–3390, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Akira Sasaki, Junta Mizuno, Naoaki Okazaki, and Kentaro Inui. 2016. Stance classification by recognizing related events about targets. In *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 582–587. IEEE.

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

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2021. Stance detection benchmark: How robust is your stance detection? *KI-Künstliche Intelligenz*, pages 1–13.

Chen Shi, Shujie Liu, Shuo Ren, Shi Feng, Mu Li, Ming Zhou, Xu Sun, and Houfeng Wang. 2016. Knowledge-based semantic embedding for machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2245–2254, Berlin, Germany. Association for Computational Linguistics.

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.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First*

AAAI Conference on Artificial Intelligence, AAAI'17, page 4444–4451. AAAI Press.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. 2018. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3664–3674, Brussels, Belgium. Association for Computational Linguistics.

Terne Sasha Thorn Jakobsen, Maria Barrett, and Anders Søgaard. 2021. Spurious correlations in cross-topic argument mining. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 263–277, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Amir Pouran Ben Veyseh, Javid Ebrahimi, Dejing Dou, and Daniel Lowd. 2017. A Temporal Attentional Model for Rumor Stance Classification. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pages 2335–2338, New York, NY, USA. Association for Computing Machinery.

Marilyn Walker, Jean Fox Tree, Pranav Anand, Rob Abbott, and Joseph King. 2012. A corpus for research on deliberation and debate. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 812–817, Istanbul, Turkey. European Language Resources Association (ELRA).

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1405–1418, Online. Association for Computational Linguistics.

Penghui Wei and Wenji Mao. 2019. Modeling Transferable Topics for Cross-Target Stance Detection. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, pages 1173–1176, New York, NY, USA. Association for Computing Machinery.

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.

Bowen Zhang, Min Yang, Xutao Li, Yunming Ye, Xiaofei Xu, and Kuai Dai. 2020. Enhancing cross-target stance detection with transferable semantic-emotion knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3188–3197, Online. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

A Appendix

A.1 Experimental Details

- All models are trained using five epochs, batch size of 16, a learning rate of 0.00002, a warmup-up ratio of 0.2 with linear scheduling, and AdamW (Loshchilov and Hutter, 2019) as optimizer. The hyperparameters tuned during training are described in the main paper (see §5.2).

- We use CUDA 11.6, Python v3.8.10, torch v1.10.0, and transformers v4.13.0 as software environment and a mixture of NVIDIA P100, V100, A100, A6000 as GPU hardware.

- We load all pretrained language models from HuggingFace model hub. In detail, we use the following model tags: `bert-base-uncased` for BERT, `google/electra-base-discriminator` for ELECTRA, and `roberta-base` for RoBERTa.

- We use the captum library (v0.5.0) to calculate the vector-norms for approximating token-attributions (Kobayashi et al., 2020) in §6.3.

- We use the statsmodel library (v0.13.2) to calculate statistical significant differences using the Bhapkar test (Bhapkar, 1966) with $p < 0.05$.

- We use sklearn (Pedregosa et al., 2011) for computing evaluation metrics (e.g. macro-F1).

- We measured the average training runtime of models on the argmin dataset as a reference. `BERT+Target` and `BERT+ConceptNet` needed 618 seconds whereas `INJECT` needed 400 seconds.

- We use the seeds $[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]$.

A.2 Datasets

We provide details about the individual split proportions for the cross-target evaluation setup in Table 5. For more information on each individual dataset, we refer to Schiller et al. (2021) and Hardalov et al. (2021).

Dataset	Train	Dev	Test	Total
arc	12,382	1,851	3,559	17,792
argmin	6,845	1,568	2,726	11,139
emergent	1,638	433	524	2,595
fnc1	42,476	7,496	25,413	75,385
iac1*	4,221	453	923	5,597
ibmcs	935	104	1,355	2,394
mtsd	6,227	1,317	1,366	8,910
perspectrum	6,978	2,071	2,773	11,822
poldeb	4,753	1,151	1,230	7,134
rumor*	6,093	299	505	7,106 (10,237)
scd	3,251	624	964	4,839
semeval2016t6	2,497	417	1,249	4,163
semeval2019t7*	5,205	1,478	1,756	8,439 (8,529)
snopes	14,416	1,868	3,154	19,438
vast	13,477	2,062	3,006	18,545
wtwt	25,193	7,897	18,194	51,284

Table 5: Number of examples per data split for the cross-target evaluation setting. For datasets marked with *, not all tweets could be downloaded or we discovered empty instances which we excluded (in comparison to the numbers provided by Hardalov et al. (2021)); for mtsd, we received the full dataset by the original authors; the original number of tweets is in parentheses.

A.3 Evaluation with ideal context

To evaluate our goal of robust integration of contextual information using INJECT, we contrast both context integration approaches in a scenario with both *ideal* context, i.e. the contextual information is guaranteed to be beneficial in predicting the correct class, and random context. To showcase, we use the e-SNLI (Camburu et al., 2018) corpus for natural language inference and the Snopes (Hanselowski et al., 2019) corpus for claim verification. We use the provided explanations (e-SNLI, m=1) and evidences (Snopes, m=10) as ideal context, respectively. As random (but syntactically correct) context, we randomly extract sentences from the Gutenberg corpus[7] included in NLTK (Bird, 2006). Table 6 compares a BERT baseline without context, BERT with context integration via concatenation (`BERT`$\oplus$), and integration via INJECT (`BERT`$\otimes$).

The results demonstrate INJECT successfully leveraging the contextual information while not outperforming direct integration in the case of ideal context. However, when provided with irrelevant context, INJECT is closer to the context-free baseline performance.

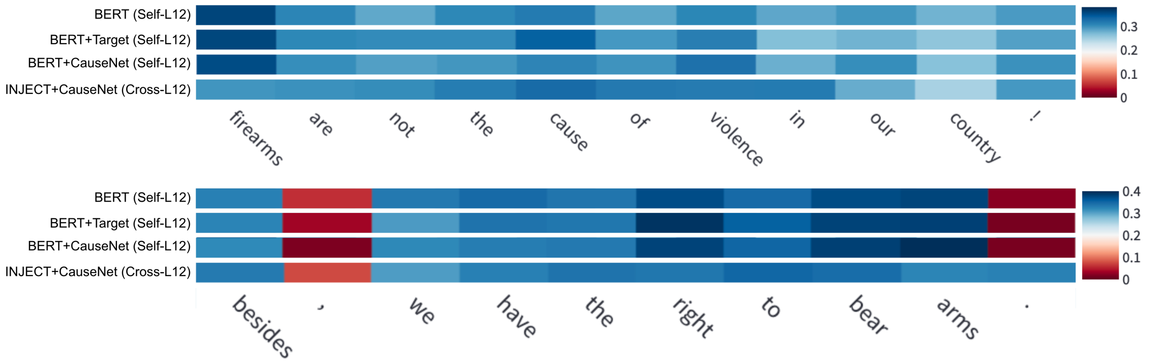

Figure 5: Two examples of the argmin dataset. The first is an argument against gun control, while the second supports it. It shows the token-level attribution for `BERT`, `BERT+Target`, `BERT+CauseNet`, and `INJECT+CauseNet`.

	e-SNLI (Ideal)	e-SNLI (Random)	Snopes (Ideal)	Snopes (Random)
`BERT`	90.33	90.33	51.8	51.8
`BERT`⊕	98.70	90.08	78.0	49.7
`BERT`⊗	98.35	90.52	75.8	51.1

Table 6: Comparison of context integration via concatenation (`BERT`⊕) and INJECT (`BERT`⊗) on e-SNLI and Snopes. Original dataset splits are used. Scores are macro-F1 averaged across three seeds.

A.4 Identification of target-specific label correlations

We examine internal processes in the model architecture by analyzing how relevant a token is compared to how much a model attributes to the token. In detail, we calculate for the 5% most relevant tokens for target and label the correlation of this relevance and the model attribution on them.

Token Relevance We consider the probability of a token to appear in combination with a label l and target t. A higher probability indicates that a token is more likely to occur within a label-target combination.

In detail, we first calculate the relevance as the maximum log-odds-ratio $r_{(w,(l_i,t_j))}$ (Kawintiranon and Singh, 2021) over all possible combinations of labels $L = \{l_1, ..., l_n\}$ and targets $T = \{t_1, ..., t_k\}$ for a given token w. We define $o_{(w,(l,t))}$ (Equation 1) as the probability of token t appearing in combination with label l and target t, with $c(w,(l,t))$ denoting the counts of w in texts with label l and target t. Next, we calculate the maximum log odds-ratio $r_{(t,(L,T))}$ as in Equation 2. This tells us how specific a token w is at max. for a label-target combination.

$$o_{(w,(l_i,t_j))} = \frac{c(w,(l_i,t_j))}{c(\neg w,(l_i,t_j))} \tag{1}$$

$$r_{(w,(L,T))} = \max_{(l_i,t_j)\in L\times T} log(\frac{o_{(w,(l_i,t_j))}}{o_{(w,\neg(l_i,t_j))}}) \tag{2}$$

Token Attributions To approximate a token's attribution, we calculate the vector-norms (Kobayashi et al., 2020) for the output of the 12th layer.

We provide anecdotal examples in Figure 5 along with their token-level attribution of the 12th layer from (`BERT`, `BERT+Target`, `BERT+CauseNet`) and `INJECT+CauseNet`. For the first three, we use the self-attention and for the latter one the cross-attention. In the first example, `INJECT+CauseNet` made the right prediction while all `BERT`-based models failed and vice-versa for the second one. In both examples, we see lower attribution for target-specific terms like *firearms* or *arms* and higher attribution for terms with general use like *besides*, *cause*, or *to*. `INJECT+CauseNet` makes the correct prediction while `BERT+Target` failed due to its high attribution to *firearms* - an example of a spurious correlation. However, in some cases this can also lead to erroneous predictions as in the second example where `INJECT+CauseNet` gives less importance to the specific - and in this case important - tokens of the sentences (*right to bear arms*).

A.5 Dataset Characteristics

In Table 7, we provide relevant dataset characteristics for each dataset in the stance detection benchmark. To compute label imbalance, we first calculate the mean and standard deviation of the number of instances per label. The label imbalance is then

[7] `http://www.gutenberg.org/`

	arc	iac1	perspectrum	poldeb	scd	emergent	fnc1	snopes	mtsd	rumor	semeval16	semeval19	wtwt	argmin	ibmcs	vast
Number of Labels	4	3	2	2	2	3	4	2	3	5	3	4	4	2	2	2
Label Imbalance	1,16	0,55	0,03	0,11	0,20	0,41	1,13	0,48	0,23	0,61	0,37	1,09	0,58	0,11	0,11	0,22
Train-Test-ratio	3.48	4.57	2.52	3.86	3.37	3.13	1.67	4.57	4.56	12.48	2	2.96	1.38	2.51	0.69	4.48
Train-Test-Vocabulary Overlap	16461	16939	2875	7553	5171	861	11244	8092	4876	818	2666	3248	7836	4454	1361	6271
FRES Mean	63.07	70.27	53.12	65.11	70.70	66.37	61.44	61.36	71.43	58.08	67.43	58.62	48.41	51.73	39.94	63.19
FRES St.Dev.	14.2	13.6	29.6	37.4	29.5	21.5	10.3	26.3	17.5	57.7	22.2	49.3	26.8	22.8	29.4	14.1
Baseline St.Dev.	0.8	2.9	0.8	2.8	1.7	1.5	1.3	0.8	2.1	9.8	0.6	2.6	4.1	1.4	1.5	1

Table 7: Overview of the dataset-characteristic for each dataset.

defined as the division of the standard deviation by the mean.

A.6 Knowledge

The information about the average length of the retrieved contextual knowledge is given in Table 8. We observe substantially longer paragraphs extracted from CauseNet which is not surprising as CauseNet consists of passages extracted from Wikipedia.

A.6.1 CauseNet

We ignore concepts which are shorter than 3 characters or consist of one of the following modal verbs ("must", "shall", "will", "should", "would", "can", "could", "may", "might").

A.6.2 Prompts

We manually evaluated the following prompts for both single and combination inputs. As reported in related work (Jiang et al., 2020; Schick and Schütze, 2021), the generated text is sensible to wording and punctuation in the prompt. We made similar experiences and removed all punctuation at the end of the prompt to prevent the model from generating outputs of short length.

A.7 On Efficiency of INJECT

From an efficiency point-of-view, INJECTprocesses a text and corresponding contexts more efficiently than via SEP integration. This is because there is no self-attention over *input* and *context* jointly where the attention dimension is $d_{sep} = (\text{len}(input) + \text{len}(context)) \times (\text{len}(input) + \text{len}(context))$. For INJECT, in contrast, input and context are processed in separate encoders with attention dimensions $d_{input} = \text{len}(input) \times \text{len}(input)$ and $d_{context} = \text{len}(context) \times \text{len}(context)$ on every layer. Just in the INJECT-layer, there are two additional attention blocks with dimensions $d_{cross\ context} = \text{len}(input) \times \text{len}(context)$ and $d_{cross\ input} = \text{len}(context) \times \text{len}(input)$.

Knowledge Source	arc	iac1	perspectrum	poldeb	scd	emergent	fnc1	snopes	mtsd	rumor	semeval16	semeval19	wtwt	argmin	ibmcs	vast
`ConceptNet`	5.1	5.1	5.5	5.1	5.2	5.5	5.1	5.5	5.6	6.0	5.5	5.6	5.6	5.3	5.3	5.1
`CauseNet`	91.2	112.1	20.5	78.4	69.8	34.1	137.6	40.3	56.4	50.6	52.1	47.0	43.0	36.4	23.7	89.5
`T0pp-NP`	13.1	13.1	13.0	12.9	12.5	13.3	14.1	13.6	12.9	13.1	12.4	13.0	12.5	12.5	13.5	13.1
`T0pp-NP-T`	9.9	12.7	10.5	12.1	11.9	11.6	16.7	11.9	14.0	13.6	12.7	9.4	11.1	12.7	11.7	12.2

Table 8: Average length for each combination of knowledge extraction method and dataset.

Prompt	**Usage**
define a	✓
what is a	
describe a	
what is the definition of a	✓
explain a	✓
relation between a and b	✓
how is a related to b	✓
explain a in terms of b	✓

Table 9: Prompts which have been evaluated for generating contextual knowledge for stance detection.