FadeIT at EVALITA 2026: Overview of the Fallacy Detection in Italian Social Media Texts Task

Alan Ramponi[1,*], Sara Tonelli[1]

[1]Fondazione Bruno Kessler, Digital Humanities Unit – Trento, Italy

Abstract

FadeIT is the first shared task on fallacy detection in social media texts in Italian, an understudied language for this task. FadeIT relies on Faina, a fallacy detection dataset that includes span-level annotations with overlaps for 20 fallacy types in social media texts about migration, climate change, and public health over a 4-year time period. The shared task is articulated into two subtasks at different granularities: i) *post-level fallacy detection*, aiming at predicting the fallacy types expressed in each input post, and ii) *span-level fallacy detection*, aiming at predicting all text segments expressing any given fallacy type in each input post. Participants' systems are evaluated against two equally valid gold standards (i.e., parallel annotations in Faina) to account for natural disagreement, in line with recent work advocating the importance of considering human label variation in subjective tasks. FadeIT has attracted wide interest at Evalita 2026 with a total of 25 runs submitted by 7 participant teams. In this paper, we present the task setup, including the data used and the evaluation criteria, as well as the results obtained by all participant teams, an analysis of their approaches, and insights for future research on the topic.

Keywords

Natural language processing, fallacy detection, argumentation mining, human label variation

1. Introduction

Fallacies are arguments that seem valid but are not [1, 2]; namely, statements that are logically flawed or in which evidence is replaced by emotional cues. Fallacious argumentation frequently occurs in everyday discourse, either intentionally – for persuading the audience – or unintentionally. With the widespread use of online platforms, fallacious social media posts have the potential to mislead a large audience, in some cases leading to the proliferation of misinformation [3]. Therefore, recognizing fallacies is paramount not only to limit the spread of misleading content, but also to develop individuals' critical thinking skills and promote democratic debate [4].

Motivated by the intrinsic difficulty of the fallacy detection task for automated systems including large language models (LLMs) [5, 6], we present FadeIT, the first shared task on fallacy detection in Italian social media texts. FadeIT has been organized as part of the Evalita 2026 evaluation campaign [7], and comprises two subtasks of increasing complexity: a subtask that deals with the detection of fallacies at the post level (i.e., a multi-label text classification problem) and a more challenging subtask requiring the detection of fallacies at the span level (i.e., a multi-label span classification problem). FadeIT relies on Faina [6], the first dataset for fallacy detection in Italian embracing multiple plausible answers and natural disagreement, with annotations across an inventory of 20 fallacy types at the fine-grained level of text segments with potential overlaps. It covers public discourse on migration, climate change, and public health issues in social media posts over a large time frame of 4 years. The evaluation of participants' systems is carried out by comparing their submitted predictions with multiple gold standards – included in the Faina dataset as parallel annotations – in order to account for human label variation [8] – i.e., the genuine disagreement that naturally occurs in subjective NLP tasks.

In the following sections, we present details on the task (Section 2), the data used for the competition (Section 3), the evaluation setup (Section 4), the participant teams and their results (Section 5), followed by an analysis and discussion (Section 6) and our conclusions (Section 7).

EVALITA 2026: 9th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Feb 26 – 27, Bari, IT

*Corresponding author.

alramponi@fbk.eu (A. Ramponi); satonelli@fbk.eu (S. Tonelli)

0000-0002-4305-2404 (A. Ramponi); 0000-0001-8010-6689 (S. Tonelli)

2. Task description

The FadeIT shared task focuses on detecting fallacies expressed in Italian social media texts about migration, climate change, and public health issues at different granularities: at the post level (subtask A; Section 2.1) and at the span level (subtask B; Section 2.2). For each post, there can be zero, one, or more fallacies, among an inventory of 20 fallacy types, to be detected at the chosen granularity. We refer to Section 3.2 for the list of fallacy types and to the Faina dataset [6] for detailed descriptions.

2.1. Subtask A: Post-level fallacy detection

Given the text of a social media post, predict all the fallacy types expressed in it. This is a multi-label classification task (20 classes) and represents the easiest setup – i.e., there is no need to *locate* each occurring fallacy type within the text, just to *detect* which ones (if any) are expressed in it.

2.2. Subtask B: Span-level fallacy detection

Given the text of a social media post, predict all the text segments expressing fallacies and give each of them a fallacy type. This is a challenging, multi-label span classification task (20 classes) and represents the hardest setup – i.e., different fallacies may overlap, partially or in full, with each other. The text of the posts provided to participants is already divided into tokens.

3. Data

In this section, we summarize how the Faina dataset used for the FadeIT task has been collected (Section 3.1) and annotated (Section 3.2). Moreover, we provide information on data splits (Section 3.3) and format (Section 3.4). Further details are provided in the original paper introducing the dataset [6].

3.1. Data collection

Data collection was conducted using the Twitter APIs in February 2023.[1] We collected social media posts covering discourse on migration, climate change, and public health using a manually curated list of keywords. The time period of the posts represented in the dataset is from `2019-01-01` to `2022-12-31`. From this collection, we selected the posts with at least 5 tokens and greatest potential impact to the society – i.e., those with the highest number of retweets and likes, as in Nakov et al. [9]. We mitigated topic and temporal biases by keeping the top 10 posts for each month and topic combination (e.g., 10 posts about "migration" posted in `2021-01`). We further mitigate authors' stylistic bias by excluding multiple posts by the same user and resampling them until we obtained the 10 posts. As a result, the Faina dataset consists of 1,440 posts balanced across topics (480 per topic) and time (360 per year).

3.2. Data annotation

The collected posts underwent fine-grained annotation at the span-level with overlaps. Due to the difficulty of the task, we devised an annotation protocol that consisted of five rounds of annotation and discussion among two expert annotators ($\mathcal{A}_1$ and $\mathcal{A}_2$). At each round, each annotator individually located and classified text segments expressing fallacies. Then, annotators met to discuss the instances that diverged in the assigned fallacy type, span extent, or both. Instead of forcing a "single ground truth" in data, the goal of the discussion phase was to minimize annotation errors (e.g., due to attention drops) whilst keeping signals of human label variation (e.g., genuine disagreement, such as multiple plausible annotations due to different interpretations of the text). Overall, Faina consists of 11,064 annotated spans ($5{,}532_{\pm 253}$/annotator) across 58,490 tokens. For allowing a post-level version of the task to be addressed (i.e., subtask A), span-level annotations have also been transposed to the post-level.[2]

[1] At that time, the Twitter (now X) APIs for research purposes were still available for free.
[2] This was done by assigning to each post the set of unique fallacy span types occurring in it.

Inventory of fallacy types The 20 fallacy categories are: *Ad hominem* (AH), *Appeal to authority* (AA), *Appeal to emotion* (AE), *Causal oversimplification* (CO), *Cherry picking* (CP), *Circular reasoning* (CR), *Doubt* (DO), *Evading the burden of proof* (EP), *False analogy* (FA), *False dilemma* (FD), *Flag waving* (FW), *Hasty generalization* (HG), *Loaded language* (LL), *Name calling or labeling* (NC), *Red herring* (RH), *Slippery slope* (SS), *Slogan* (SL), *Strawman* (ST), *Thought-terminating cliché* (TC), and *Vagueness* (VA). For fallacy definitions, inter-annotator agreement scores before and after discussions, annotation guidelines, and statistics, please refer to the paper introducing FAINA [6]. We provide examples from FAINA in Figure 1.

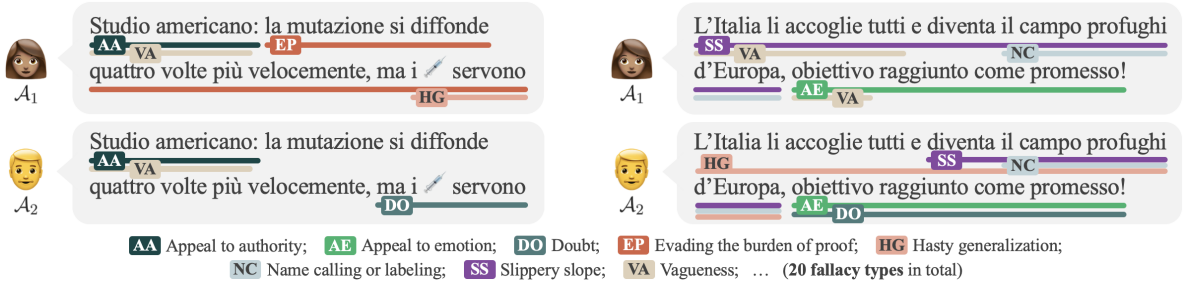

Figure 1: Example of annotated posts from the FAINA dataset [6], keeping genuine disagreement among two expert annotators ($\mathcal{A}_1$, $\mathcal{A}_2$). English translations: "*American study: mutation spreads four times faster, but are needed*" (*left*); "*Italy welcomes them all and becomes Europe's refugee camp, goal achieved as promised!*" (*right*).

3.3. Data splits

For the purpose of the shared task, the dataset has been split into two official sets: one for training/development (80%; 1,152 posts)[3] and one for testing (20%; 288 posts). These have been created by paying particular attention to label, time, and topic distribution across the splits to ensure reliability in the official evaluation. The posts represented in the splits are the same for both subtask A and B.

3.4. Data format

Subtask A For *post-level fallacy detection*, the data is in a tab-separated format with a header line. Each line consists of information about each post (i.e., id, date, topic, text, labels). Post-level annotations by each annotator are provided in separate columns and multiple annotations for the same post and annotator are separated by a pipe. Specifically, each post is represented as shown in Table 1 (*top*).

Subtask B For *span-level fallacy detection*, the data format is based on the CoNLL format. Each post is separated by a blank line and consists of a header with post information, followed by each token in the text (with tab-separated information) separated by newlines. Token annotations follow the BIO scheme (i.e., B: begin, I: inside, O: outside) and multiple annotations for the same token and annotator are separated by a pipe. Specifically, a post in FAINA is represented as shown in Table 1 (*bottom*).

Table 1
FAINA format for subtask A (*top*) and subtask B (*bottom*). Each variable is described in Section 3.4.

$POST_ID	$POST_DATE	$POST_TOPIC	$POST_TEXT	$LABELS_BY_ANN_1	$LABELS_BY_ANN_2

```
# post_id = $POST_ID
# post_date = $POST_DATE
# post_topic_keywords = $POST_TOPIC
# post_text = $POST_TEXT
$TOKEN_1    $TOKEN_1_TEXT    $TOKEN_1_LABELS_BY_ANN_1    $TOKEN_1_LABELS_BY_ANN_2
...
$TOKEN_N    $TOKEN_N_TEXT    $TOKEN_N_LABELS_BY_ANN_1    $TOKEN_N_LABELS_BY_ANN_2
```

[3]Participant teams have been left free to decide how to split the training/development set to tune and select their systems.

The variables in Table 1 are defined as follows:

- **$POST_ID**: the identifier of the post, different from the Twitter one to preserve user's anonymity;
- **$POST_DATE**: the date of the post (in the YYYY-MM format);
- **$POST_TOPIC**: the topic of the post (i.e., "migration", "climate change", or "public health");
- **$POST_TEXT**: the text of the post, anonymized with placeholders;[4]
- **$LABELS_BY_ANN_j**: the fallacy label(s) assigned by annotator *j* for the post (e.g., "Vagueness", "Strawman"). In the case where multiple labels for the post are assigned by the same annotator *j*, these are separated by a pipe and ordered lexicographically, e.g., "Strawman|Vagueness". In the case where no labels for the post are assigned by the same annotator *j*, the label is empty;
- **$TOKEN_i**: the index of the token within the post (i.e., an incremental integer);
- **$TOKEN_i_TEXT**: the text of the *i*-th token within the post;
- **$TOKEN_i_LABELS_BY_ANN_j**: the fallacy label(s) assigned by annotator *j* for the *i*-th token within the post. Each label follows the format $BIO-$LABEL, where $BIO is the BIO tag and $LABEL is the fallacy label (e.g., "Vagueness", "Strawman"), e.g., "B-Vagueness", "I-Strawman", and "O". In the case where multiple labels for the *i*-th token are assigned by the same annotator *j*, these are separated by a pipe and ordered lexicographically by $LABEL, e.g., "I-Strawman|B-Vagueness".

We left it up to participant teams to decide whether to aggregate gold annotations by different annotators (e.g., using majority voting), using only one, or leveraging all of them for designing their systems. Nevertheless, to account for human label variation, systems are evaluated against all gold standards (Section 4.1). An example of a post following the aforementioned data formats is presented in Table 2.

Table 2
Post from Figure 1 (*left*) following the data format for subtask A (*top*) and B (*bottom*). In both cases, the last two columns indicate annotations provided by annotators $\mathcal{A}_1$ and $\mathcal{A}_2$ due to different (equally valid) interpretations.

658	2021-06	public health	Studio americano: la mutazione si diffonde quattro volte più velocemente, ma i 🪰 servono	Appeal-to-authority|Evading-the-burden-of-proof|Hasty-generalization|Vagueness	Appeal-to-authority|Doubt|Vagueness

```
# post_id = 658
# post_date = 2021-06
# post_topic_keywords = public health
# post_text = Studio americano: la mutazione si diffonde quattro volte più velocemente, ma i 🪰 servono
1     Studio       B-Appeal-to-authority|B-Vagueness                                  B-Appeal-to-authority|B-Vagueness
2     americano    I-Appeal-to-authority|I-Vagueness                                  I-Appeal-to-authority|I-Vagueness
3     :            I-Appeal-to-authority                                              I-Appeal-to-authority
4     la           B-Evading-the-burden-of-proof                                      O
5     mutazione    I-Evading-the-burden-of-proof                                      O
6     si           I-Evading-the-burden-of-proof                                      O
7     diffonde     I-Evading-the-burden-of-proof                                      O
8     quattro      I-Evading-the-burden-of-proof                                      O
9     volte        I-Evading-the-burden-of-proof                                      O
10    più          I-Evading-the-burden-of-proof                                      O
11    velocemente  I-Evading-the-burden-of-proof                                      O
12    ,            I-Evading-the-burden-of-proof                                      O
13    ma           I-Evading-the-burden-of-proof                                      B-Doubt
14    i            I-Evading-the-burden-of-proof|B-Hasty-generalization               I-Doubt
15    🪰           I-Evading-the-burden-of-proof|I-Hasty-generalization               I-Doubt
16    servono      I-Evading-the-burden-of-proof|I-Hasty-generalization               I-Doubt
```

4. Evaluation

Each team was allowed to submit up to 3 runs (i.e., predictions on the test set) for each subtask. We here introduce the metrics used for assessing performance (Section 4.1) and our baselines (Section 4.2).

4.1. Metrics

We employ different metrics for evaluating participants' runs in subtask A and B, as detailed below.

[4]User mentions, URLs, email addresses, and phone numbers are replaced with [USER], [URL], [EMAIL], and [PHONE] placeholders, respectively.

Subtask A The submitted runs are evaluated using micro- and macro-averaged precision, recall, and F_1 score, averaged on the two equally-valid gold standard annotations of the 20% held-out test set. Runs are then ranked by micro F_1 score.

Subtask B We evaluate the runs using metrics designed for span-level annotations with potential overlaps, averaged on the two equally-valid gold standard annotations of the 20% held-out test set. We adopt micro- and macro-averaged precision, recall, and F_1 score variants proposed by Da San Martino et al. [10], extended to work at the token level. Partial credit is therefore given to partial span matches, proportional to the length of the match in terms of tokens. To account for the severity of labeling errors (e.g., predicting *Red herring* instead of *Appeal to authority* is less problematic than predicting *False dilemma*), results are also computed in a "soft" evaluation mode, namely giving partial credit (i.e., 0.5 instead of 1.0) if the predicted label is an immediate parent of the actual label in the taxonomy of fallacy types by Ramponi et al. [6]. Runs are then ranked by micro F_1 score in the *soft* evaluation mode.

4.2. Baselines

As baseline systems, we provided two encoder-based models for each subtask. These models have been previously described in the paper introducing FAINA [6] and are summarized in the following.[5]

MVML-ALB and MVML-UMB models A multi-view, multi-label (MVML) model that relies on a shared encoder (either AlBERTo [11] or UmBERTo [12], i.e., ALB or UMB), uses $D = |A|$ decoders (one for each annotation view, i.e., for the labels assigned by each annotator), and outputs D sets of predicted labels containing all fallacy labels that exceed a threshold τ (with $\tau = 0.7$).

MVMD-ALB and MVMD-UMB models A multi-view, multi-decoder (MVMD) model that relies on a shared encoder (either AlBERTo [11] or UmBERTo [12], i.e., ALB or UMB), uses a separate decoder for each annotation view A and fallacy type F (i.e., $D = |A \times F|$), and outputs D sets of predicted labels (i.e., either 'B', 'I', or 'O' for each fallacy label and annotation view). All decoders are given equal importance in the computation of the multi-task learning loss.

5. Participants and results

The FADEIT shared task has attracted a total of 25 runs by 7 participant teams. Specifically, for subtask A we received 16 runs by 6 teams,[6] whereas for the more challenging subtask B we received 9 runs by 3 teams. Overall, FADEIT has been one of the most participated shared tasks at Evalita 2026 and attracted interest of teams from both academia and industry, representing institutions across five different countries (i.e., India, Italy, Japan, Netherlands, and Vietnam). An overview of participant teams' approaches is provided in Section 5.1, whereas results for both subtasks are presented in Section 5.2.

5.1. Overview of participant teams' approaches

A summary of the approaches adopted by each team is provided in the following, alphabetically ordered by team name. For additional details on each submitted run (e.g., model versions, hyper-parameter choices, data and prompt variations), we refer the reader to the system description paper of each team.

Kenji-Endo [13] The team proposed a model with focus on efficiency to tackle multiple Evalita tasks, including FADEIT's subtask A. The system is based on a decoder-only causal language model that was first pretrained on a mixture of Italian corpora, and then fine-tuned on FAINA data in a discriminative setting by taking into account class imbalance in the loss function. The team submitted a single run.

[5] The code for the baselines is available in the original FAINA's repository: https://github.com/dhfbk/faina.
[6] These include a late run by the team *MALTO* (run 3).

Label [14] The team participated in subtask A and experimented with a two-step prompting approach using LLMs – namely, by first generating a text discussing the potential presence of each fallacy type and then producing a score indicating the likelihood of each of them being present, based on both the generated analysis and the original text of the post. The scores are either used to determine if a fallacy type has to be outputted (based on a tuned threshold value; run 3) or as features, together with topic information and statistical features about the distribution of scores across fallacy types, for training a multi-layer perceptron that models and predicts individual annotators' labels (run 1 and 2).

MALTO [15] The team participated in subtask A by fine-tuning an encoder-based model pretrained on Italian data using a global multi-label classification threshold. The submitted runs reflect different hyper-parameter configurations and data composition for fine-tuning. Specifically, in run 2, the team used a binary cross-entropy loss without class weights, whereas in run 3, they used a binary cross-entropy loss with positive class weights and paraphrase-based augmented data for fine-tuning.[7]

PuDy [16] The team participated in subtask B. They adopted a span-based modeling approach using an encoder-based model and representing spans of up to 20 tokens as the concatenation of boundary span embeddings and a learned span length embedding. The system employs a hierarchy-based label propagation strategy, making fallacy types that are immediate parents of the gold fallacy sub-types to receive supervision. Moreover, the surrounding context of fallacious spans is perturbed using an LLM (in a 2-shot setting) to reduce overfitting to contextual lexical cues. The three runs reflect different hyper-parameter and data configurations of the same system.

RBG-AI [17] The team participated in both subtasks with three runs each using a unified prompt-based framework. They tested instruction-tuned LLMs in few-shot settings (namely, using 3-, 5-, and 10-shots), selecting examples that maximize diversity in terms of fallacy types, multi-label instances, and annotation views. During prediction, outputs are bounded to coarse-grained groups of fallacy types derived from semantic closeness and corpus-level distributional statistics of FAINA's fallacy inventory.[8]

TiGRO [18] The team submitted three runs for each of the subtasks. For subtask A, they experimented with a one-vs-rest strategy by fine-tuning 20 binary classifiers – one per fallacy type – using a multilingual encoder (run 1), and with a multi-task learning approach, using an encoder pretrained on Italian data and a decoder per fallacy type, either considering post-level annotations (run 2) or jointly accounting for post- and span-level labels using a total of 40 decoders (run 3). For subtask B, the team employed a multi-task learning approach with 20 span-level decoders and an encoder pretrained on Italian data (run 1), employed the same system used for run 3 in subtask A (run 2), and experimented with a variant of this system by substituting the backbone encoder with a multilingual one (run 3).

UNICA [19] The team participated in subtask A and proposed different approaches based on fine-tuning and retrieval-augmented generation (RAG). All approaches used training data that was previously augmented via LLM prompting and back-translation. In run 1, a closed-weight LLM from the OpenAI family was instructed with few-shot examples. The examples were dynamically selected based on their semantic similarity with the input text and through RAG, using a closed-weight text embedding model from the same family. In run 2 and 3, fine-tuning of LLMs was conducted using models from the Gemma and Mixtral families, respectively.

5.2. Results

In this section, we provide the results on the official test set for all runs submitted by participant teams in the post-level (Section 5.2.1) and span-level (Section 5.2.2) fallacy detection setups.

[7] At the time of writing, we do not have details about run 1; we refer the reader to the MALTO's technical report.
[8] At the time of writing, we do not have details on the exact approach adopted in each run; we refer to the RBG-AI's report.

5.2.1. Subtask A: Post-level fallacy detection

The results on the test set for all the runs submitted by teams participating in subtask A are reported in Table 3. By looking at micro-averaged F_1 scores, the *TiGRO* team achieves the best results on the task (56.39 micro F_1; run 3) by modeling post- and span-level annotations jointly in a multi-task learning framework. *MALTO* follows it with a system that uses data augmentation via paraphrasing and takes into account class imbalance in the loss computation (54.63 micro F_1; run 2). The other runs by *TiGRO* place third (52.41 micro F_1; run 2) and fifth (48.65 micro F_1; run 1), with a multi-task learning approach considering post-level annotations and by employing a one-vs-rest strategy, respectively. Interestingly, all these systems do not rely on decoder-based LLMs but on encoder-based models, confirming what has been observed in previous work about the challenges of this task for LLMs [5, 6]. The submitted run that ranks the highest among those using LLMs is run 1 by *UNICA* (49.71 micro F_1). It places fourth by instructing models from the OpenAI family in a few-shot manner, using examples that were dynamically selected using semantic similarity and retrieval-augmented generation methods.

Table 3
Test set results for subtask A (*post-level fallacy detection*). P: Precision, R: Recall, F_1: F_1 score. All scores are reported in their micro- and macro-averaged flavors. Runs are ranked by decreasing micro-averaged F_1 score, and the best score for each metric is in bold. Baselines are highlighted in yellow. '*' indicates late submissions.

			micro-averaged			macro-averaged		
	Team	**Run**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
1	TiGRO	3	53.24	59.99	**56.39**	34.47	34.74	33.35
2	MALTO	2	55.75	53.60	54.63	41.95	30.95	32.63
3	TiGRO	2	53.43	51.48	52.41	35.91	27.21	27.94
4	UNICA	1	55.22	45.23	49.71	**47.13**	37.28	**36.75**
5	TiGRO	1	**62.52**	39.85	48.65	27.46	18.11	20.57
6	UNICA	3	51.19	44.26	47.45	35.08	24.95	26.14
7	Kenji-Endo	1	49.38	45.55	47.37	14.25	17.17	13.71
8	UNICA	2	58.70	38.44	46.44	35.68	19.53	22.76
9	MALTO	1	46.56	43.65	45.05	28.11	23.33	23.17
	MVML-ALB		*64.29*	*34.41*	*44.82*	*37.80*	*15.42*	*19.68*
10	RBG-AI	2	33.09	57.78	42.07	26.54	46.21	29.32
*	MALTO	3	37.45	45.04	40.88	23.53	30.02	25.64
11	RBG-AI	3	30.65	57.62	40.00	31.08	54.90	31.31
12	Label	3	27.60	**68.08**	39.26	22.01	**56.97**	29.04
13	Label	1	52.82	30.94	38.96	14.50	10.13	10.31
14	RBG-AI	1	36.35	41.11	38.57	32.77	29.60	23.42
15	Label	2	52.76	30.11	38.32	14.62	10.17	10.82
	MVML-UMB		*38.53*	*14.28*	*20.84*	*15.13*	*3.45*	*5.10*

In contrast, when looking at macro-averaged scores, *UNICA* ranks first (36.75 macro F_1; run 1). By looking at the per-fallacy scores (Figure 2), *UNICA* (run 1) obtains more balanced scores across fallacy types compared to other teams that ranked higher according to the micro F_1 metric. The high macro F_1 score obtained by the system could therefore be attributed to the good performance achieved on fallacy types that are under-represented in the data (e.g., *Causal oversimplification*, *Slippery slope*), which fine-tuned encoder-based models typically struggle to capture due to the limited number of instances available for training. Nevertheless, the winning *TiGRO* run and the run by *MALTO* that ranked second still take the second (33.35 macro F_1; run 3) and the third (32.63 macro F_1; run 2) place when looking at macro F_1 scores, respectively, indicating high robustness and showing good performance for most under-represented classes (see Figure 2). Finally, the competitive performance of *RBG-AI* (31.31 and 29.32 macro F_1; run 3 and 2) and *Label* (29.04 macro F_1; run 3) in terms of macro-averaged scores, despite placing eleventh, tenth, and twelfth according to micro F_1, respectively, is likewise attributable to the good performance on minority fallacy categories and little performance degradation on majority labels

such as *Loaded language* and *Name calling or labeling*. All runs outperform the MVML-UMB baseline, whereas MVML-ALB is still competitive, especially when looking at micro-averaged scores.

Further details on per-class scores obtained by participant teams' runs can be found in Figure 2.

Figure 2: Test set results divided by fallacy type for subtask A (*post-level fallacy detection*) in terms of F_1 score. Participant teams (with run numbers within parentheses) are on the rows, and fallacy types are on the columns. AH: Ad hominem; AA: Appeal to authority; AE: Appeal to emotion; CO: Causal oversimplification; CP: Cherry picking; CR: Circular reasoning; DO: Doubt; EP: Evading the burden of proof; FA: False analogy; FD: False dilemma; FW: Flag waving; HG: Hasty generalization; LL: Loaded language; NC: Name calling or labeling; RH: Red herring; SS: Slippery slope; SL: Slogan; ST: Strawman; TC: Thought-terminating cliché; VA: Vagueness.

5.2.2. Subtask B: Span-level fallacy detection

Test set results for all runs submitted for subtask B are shown in Table 4. All teams outperform the MVMD-UMB baseline, but only the *TiGRO* team achieves higher performance than MVMD-ALB in all runs in the *strict* mode. According to the official shared task metric (micro-averaged span-level F_1 score in the *soft* mode), all the runs by the *PuDy* team ranked first, followed by those by *TiGRO* and those by *RBG-AI*. Specifically, the best system (50.92 micro F_1, *soft*; *PuDy*, run 1) uses a span-based modeling approach and relies on the UmBERTo encoder-based model. By looking at micro-averaged scores in the *strict* mode (i.e., when requiring the prediction of the exact fallacy types, without granting partial scores for non-severe errors, see Section 4.1), we observe that the best system is the multi-task model with 40 decoders and mmBERT as encoder by *TiGRO* (42.13 micro F_1, *strict*; run 3). According to this metric, this *TiGRO* system outperforms the scores of the best *PuDy* system (31.97 micro F_1, *strict*; run 1). When looking at macro-averaged scores, *TiGRO* achieves the best results (26.05 macro F_1, *strict*; run 1), as also shown by individual fallacy scores in Figure 3. Overall, we observe that systems by both *PuDy* and *TiGRO* are competitive and can be used for different use cases. For instance, if we have no strict

requirements on the identification of exact fallacy categories, *PuDy*'s one would be the system to go. If we are interested in recognizing precise fallacy types, one would prefer the *TiGRO* system instead.

More in general, the choice of a system often depends on the fallacy types of interest. Further details on the scores for each fallacy type obtained by participant teams' runs can be found in Figure 3.

Table 4
Test set results for subtask B (*span-level fallacy detection*). P: Precision, R: Recall, F_1: F_1 score (span-level variants). All scores are reported in both *strict* and *soft* evaluation modes in their micro- and macro-averaged flavors, where applicable. Runs are ranked by decreasing micro-averaged F_1 score in the *soft* evaluation mode, and the best score for each metric is in bold. Baselines are highlighted in yellow.

			STRICT MODE						SOFT MODE		
			micro-averaged			macro-averaged			micro-averaged		
	Team	**Run**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
1	PuDy	1	27.68	37.83	31.97	26.24	22.39	21.11	43.76	60.88	**50.92**
2	PuDy	2	25.96	**40.80**	31.73	19.24	23.73	18.95	40.30	**64.79**	49.69
3	PuDy	3	29.90	32.96	31.36	20.99	18.83	18.20	45.75	52.36	48.83
4	TiGRO	3	**47.82**	37.67	**42.13**	**35.69**	23.23	25.79	**51.01**	40.25	44.98
5	TiGRO	1	38.33	40.35	39.30	31.52	**25.30**	**26.05**	42.50	45.20	43.80
6	TiGRO	2	38.23	40.05	39.11	29.85	24.16	24.76	42.68	44.87	43.74
	MVMD-ALB		*48.83*	*26.87*	*34.66*	*36.13*	*16.42*	*20.87*	*52.98*	*29.48*	*37.89*
7	RBG-AI	3	19.47	25.25	21.99	17.68	16.43	15.18	24.18	32.44	27.71
8	RBG-AI	2	19.21	18.24	18.71	17.52	12.98	12.66	24.67	23.96	24.31
9	RBG-AI	1	17.13	11.01	13.41	16.66	7.67	9.55	20.70	13.27	16.17
	MVMD-UMB		*60.94*	*3.05*	*5.80*	*10.51*	*3.21*	*3.80*	*65.97*	*3.28*	*6.25*

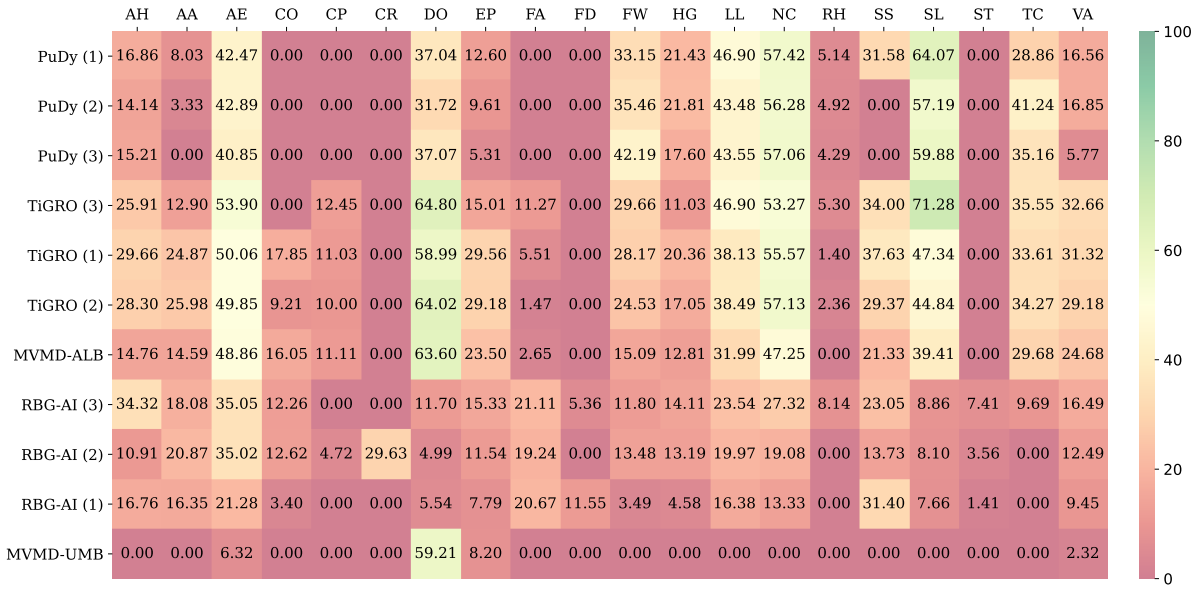

Figure 3: Test set results divided by fallacy type for subtask B (*span-level fallacy detection*) in terms of F_1 score (span-level variant, *strict* evaluation mode). Participant teams (with run numbers within parentheses) are on the rows, and fallacy types are on the columns. AH: Ad hominem; AA: Appeal to authority; AE: Appeal to emotion; CO: Causal oversimplification; CP: Cherry picking; CR: Circular reasoning; DO: Doubt; EP: Evading the burden of proof; FA: False analogy; FD: False dilemma; FW: Flag waving; HG: Hasty generalization; LL: Loaded language; NC: Name calling or labeling; RH: Red herring; SS: Slippery slope; SL: Slogan; ST: Strawman; TC: Thought-terminating cliché; VA: Vagueness.

6. Analysis and discussion

Models All participant teams used transformer-based language models as part of their systems. Among encoder-based models, *MALTO* used AlBERTo [11], *PuDy* employed UmBERTo [12], and *TiGRO* used both AlBERTo and mmBERT [20] in their runs. As regards decoder-based models, open-weight LLMs such as Gemma 3 12B [21] and LLaMa 3.1 8B [22] have been used by *Label* and *RBG-AI* teams, respectively, as well as Mixtral 8x7B [23] and Gemma 3 12B [21] by *UNICA*. Closed-weight models have been used by *UNICA* (i.e., GPT-5, GPT-5.1, and text-embedding-3-small), *PuDy* employed Gemini [24] for the contextual enrichment phase of their systems, whereas *MALTO* used ChatGPT for paraphrase-based data augmentation. *TiGRO* is the only team that used multi-task learning in their runs by leveraging the MaChAmp toolkit [25], showing improvements in performance compared to a single task setup. Finally, the *Kenji-Endo* team trained a causal language model based on the Qwen3 architecture [26]. Given the different strengths of encoder- and decoder-based models in the task (Section 5.2), studying the interplay among them is a valuable direction for future work.

Human label variation and extra-linguistic information Although we provide the training/development set with parallel annotators' labels as well as topic and time period metadata for each post, this information has not been extensively leveraged by participant teams. The only exception is the *Label* team, that explicitly used the genuine disagreement in the FAINA data for training their classifier and predicting annotator-specific fallacy labels. They also experimented by using topic information, demonstrating that it is a useful feature for fallacy classification. We expect that future work will explore more approaches in this direction, embracing both human label variation [8] and addressing out-of-distribution generalization [27] by leveraging topic and time period information in FAINA data.

Data augmentation Three teams used data augmentation strategies: *MALTO*, *PuDy*, and *UNICA*. Specifically, *MALTO* (run 3) employed ChatGPT to generate paraphrases of selected training data instances (i.e., those with fallacy labels appearing < 100 times in the training set annotations by $\mathcal{A}_1$) by preserving the same post-level labels. Along with original data instances, they then used augmented posts to fine-tune their model for run 3. *PuDy* used Gemini for perturbing the context of the span (i.e., the text before and after it) to be classified. While *PuDy*'s approach seems promising for our task, *MALTO*'s one led to performance degradation. As noted by *MALTO*, their approach is likely to introduce label noise, affecting learning. To increase the chance that labels associated to generated paraphrases are still applicable, in future work a classifier trained on gold data can be used for further validation, as previously done for other tasks such as hate speech detection [28, 29]. Finally, all runs by *UNICA* used augmented training data obtained by instructing GPT-5.1 to generate additional examples for minority fallacy types (i.e., 50 posts for each fallacy type appearing $< 3\%$ of the times in the original training set). Moreover, they further augmented training data by back-translating – with Spanish and French as pivot languages – a random subset of the training data using OPUS-MT translation models [30], discarding augmented instances that exhibited a cosine similarity $< 75\%$ compared to their original counterparts.

Overall, data augmentation leads to mixed results and its efficacy depends on the specifics of each approach. Besides augmentation, future work can consider to enrich existing instances with additional layers of information to be leveraged, such as check-worthiness [31] and argumentation schemes [32].

7. Conclusions

This paper provided an overview of FADEIT, the first shared task on fallacy detection in Italian social media posts organized as part of Evalita 2026. FADEIT attracted notable interest from the research community, registering a total of 25 submitted runs by 7 participant teams from institutions across five different countries. The results of the shared task and our analysis suggest that there is still ample room for improvement in fallacy detection performance, especially for the more challenging yet analytically

useful span-level setup. We hope that our shared task, the dataset, and the evaluation protocol will foster further research in fallacy detection with human label variation.

Acknowledgments

This work has been funded by the European Union's Horizon Europe research and innovation programmes under grant agreement No. 101070190 (AI4Trust) and under the Marie Skłodowska-Curie grant agreement No. 101073351 (HYBRIDS).

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] C. L. Hamblin, Fallacies, Advanced Reasoning Forum, Socorro, USA, 2022.

[2] C. W. Tindale, Fallacies and Argument Appraisal, Critical Reasoning and Argumentation, Cambridge University Press, Cambridge, UK, 2007. doi:https://doi.org/10.1017/CBO9780511806544.

[3] E. Musi, M. Aloumpi, E. Carmi, S. Yates, K. O'Halloran, Developing fake news immunity: Fallacies as misinformation triggers during the pandemic, Online Journal of Communication and Media Technologies 12 (2022) e202217. doi:https://doi.org/10.30935/ojcmt/12083.

[4] U. Ecker, J. Roozenbeek, S. van der Linden, L. Q. Tay, J. Cook, N. Oreskes, S. Lewandowsky, Misinformation poses a bigger threat to democracy than you might think, Nature 630 (2024) 29–32. doi:https://doi.org/10.1038/d41586-024-01587-3.

[5] T. Alhindi, T. Chakrabarty, E. Musi, S. Muresan, Multitask instruction-based prompting for fallacy recognition, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 8172–8187. URL: https://aclanthology.org/2022.emnlp-main.560/. doi:10.18653/v1/2022.emnlp-main.560.

[6] A. Ramponi, A. Daffara, S. Tonelli, Fine-grained fallacy detection with human label variation, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 762–784. URL: https://aclanthology.org/2025.naacl-long.34/. doi:10.18653/v1/2025.naacl-long.34.

[7] F. Cutugno, A. Miaschi, A. P. Aprosio, G. Rambelli, L. Siciliani, M. A. Stranisci, EVALITA 2026: Overview of the 9th evaluation campaign of natural language processing and speech tools for Italian, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[8] B. Plank, The "problem" of human label variation: On ground truth in data, modeling and evaluation, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 10671–10682. URL: https://aclanthology.org/2022.emnlp-main.731/. doi:10.18653/v1/2022.emnlp-main.731.

[9] P. Nakov, A. Barrón-Cedeño, G. Da San Martino, F. Alam, R. Míguez, T. Caselli, M. Kutlu, W. Zaghouani, C. Li, S. Shaar, H. Mubarak, A. Nikolov, Y. S. Kartal, Overview of the CLEF-2022 CheckThat! lab task 1 on identifying relevant claims in tweets, in: Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2022. URL: https://ceur-ws.org/Vol-3180/paper-28.pdf.

[10] G. Da San Martino, S. Yu, A. Barrón-Cedeño, R. Petrov, P. Nakov, Fine-grained analysis of propaganda in news articles, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 5636–5646. URL: https://aclanthology.org/D19-1565/. doi:10.18653/v1/D19-1565.

[11] M. Polignano, P. Basile, M. de Gemmis, G. Semeraro, V. Basile, AlBERTo: Italian BERT language understanding model for NLP challenging tasks based on tweets, in: R. Bernardi, R. Navigli, G. Semeraro (Eds.), Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019), CEUR Workshop Proceedings, Bari, Italy, 2019, pp. 312–317. URL: https://aclanthology.org/2019.clicit-1.47/.

[12] L. Parisi, S. Francia, P. Magnani, UmBERTo: An Italian language model trained with whole word masking, https://github.com/musixmatchresearch/umberto, 2020. Accessed: 2026-01-01.

[13] C. J. Scozzaro, M. Rinaldi, G. Mittone, M. A. Stranisci, Kenji-Endo: a BabyLM @EVALITA, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[14] T. Labruna, E. Papadopulos, Label at FadeIT: Fallacy-aware LLM reasoning for score-based classification, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[15] M. Salami, L. M. Rodia, V. Schiau, E. A. Munis, C. Savelli, F. Giobergia, MALTO at FadeIT: A BERT-based system for multi-label fallacy detection in Italian social media, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[16] D. D. Phu, S. B. Hong, T. V. Dang, PuDy at FadeIT: Enhancing fine-grained fallacy detection with hierarchy-aware training and contextual enrichment, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[17] Meenakshi, J. R, R. U, B. G. HB, M. Ptaszynski, RBG-AI at FadeIT: Prompted LLMs with label abstraction for logical fallacy detection, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[18] S. Atzeni, G. Sarti, T. Caselli, M. Nissim, TiGRO at FadeIT: E pluribus unum – A multi-task approach to fallacy detection and span identification, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[19] M. Fenu, M. Atzori, Unica at FadeIT: Adapting large language models to fallacy identification in social networks, in: Proceedings of the Ninth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2026), CEUR.org, Bari, Italy, 2026.

[20] M. Marone, O. Weller, W. Fleshman, E. Yang, D. Lawrie, B. V. Durme, mmBERT: A modern multilingual encoder with annealed language learning, arXiv preprint arXiv:2509.06888 (2025). URL: https://arxiv.org/abs/2509.06888.

[21] Gemma Team, et al., Gemma 3 technical report, arXiv preprint arXiv:2503.19786 (2025). URL: https://arxiv.org/abs/2503.19786.

[22] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, et al., The Llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024). URL: https://arxiv.org/abs/2407.21783.

[23] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, et al., Mixtral of experts, arXiv preprint arXiv:2401.04088 (2024). URL: https://arxiv.org/abs/2401.04088.

[24] Gemini Team Google, et al., Gemini: A family of highly capable multimodal models, arXiv preprint arXiv:2312.11805 (2025). URL: https://arxiv.org/abs/2312.11805.

[25] R. van der Goot, A. Üstün, A. Ramponi, I. Sharaf, B. Plank, Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP, in: D. Gkatzia, D. Seddah (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Online, 2021, pp. 176–197. URL: https://aclanthology.org/2021.eacl-demos.22/. doi:10.18653/v1/2021.eacl-demos.22.

[26] A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, C. Zheng, D. Liu, F. Zhou, F. Huang, F. Hu, H. Ge, H. Wei, H. Lin, J. Tang, J. Yang, et al., Qwen3 technical report, arXiv preprint arXiv:2505.09388 (2025). URL: https://arxiv.org/abs/2505.09388.

[27] A. Ramponi, B. Plank, Neural unsupervised domain adaptation in NLP—A survey, in: D. Scott, N. Bel, C. Zong (Eds.), Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 6838–6855. URL: https://aclanthology.org/2020.coling-main.603/. doi:10.18653/v1/2020.coling-main.603.

[28] C. Casula, S. Vecellio Salto, A. Ramponi, S. Tonelli, Delving into qualitative implications of synthetic data for hate speech detection, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 19709–19726. URL: https://aclanthology.org/2024.emnlp-main.1099/. doi:10.18653/v1/2024.emnlp-main.1099.

[29] T. Wullach, A. Adler, E. Minkov, Fight fire with fire: Fine-tuning hate detectors using large samples of generated hate speech, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 4699–4705. URL: https://aclanthology.org/2021.findings-emnlp.402/. doi:10.18653/v1/2021.findings-emnlp.402.

[30] J. Tiedemann, S. Thottingal, OPUS-MT – building open translation services for the world, in: A. Martins, H. Moniz, S. Fumega, B. Martins, F. Batista, L. Coheur, C. Parra, I. Trancoso, M. Turchi, A. Bisazza, J. Moorkens, A. Guerberof, M. Nurminen, L. Marg, M. L. Forcada (Eds.), Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, European Association for Machine Translation, Lisboa, Portugal, 2020, pp. 479–480. URL: https://aclanthology.org/2020.eamt-1.61/.

[31] A. Daffara, A. Ramponi, S. Tonelli, WorthIt: Check-worthiness estimation of Italian social media posts, in: C. Bosco, E. Jezek, M. Polignano, M. Sanguinetti (Eds.), Proceedings of the Eleventh Italian Conference on Computational Linguistics (CLiC-it 2025), CEUR Workshop Proceedings, Cagliari, Italy, 2025, pp. 337–351. URL: https://aclanthology.org/2025.clicit-1.35/.

[32] P. Goffredo, M. Chaves, S. Villata, E. Cabrio, Argument-based detection and classification of fallacies in political debates, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 11101–11112. URL: https://aclanthology.org/2023.emnlp-main.684/. doi:10.18653/v1/2023.emnlp-main.684.