CoDa: Constrained Generation based Data Augmentation for Low-Resource NLP

NAACL 2024

Chandra Kiran Reddy Evuru[♠*], Sreyan Ghosh[♠*], Sonal Kumar[♠]
Ramaneswaran S[♥], Utkarsh Tyagi[♠], Dinesh Manocha[♠]
[♠]University of Maryland, College Park, [♥]NVIDIA, Bangalore, India

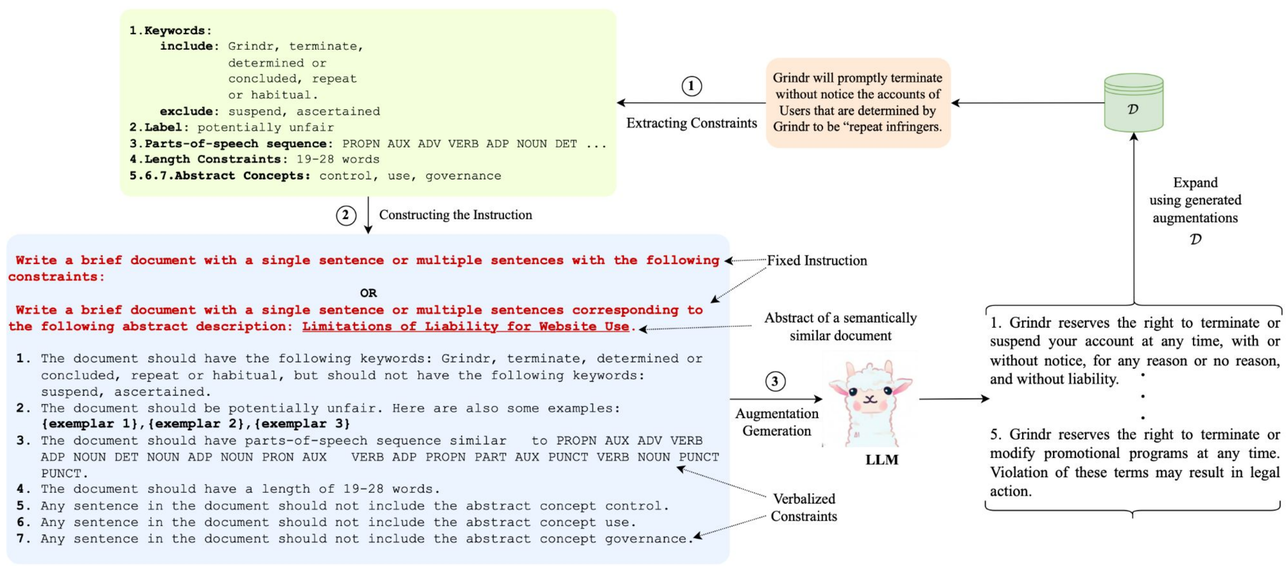
Motivation: Synthetic Data Generation is Hard!

- Given a low-resource NLU dataset, effectively generating task-specific data to expand the dataset still poses a significant challenge.
- Prior-art methods do not impose explicit controls to achieve diversity or consistency.
- Controlling LLM autoregressive generation is complex and prompting-based methods often use manual human efforts for extracting data attributes that promote consistency.

Main Contributions

1. We propose CoDa, a novel and effective constraint-based data augmentation methodology for low-resource NLP.
2. CoDa works with any off-the- shelf instruction-tuned LLM in a training-free fashion and provides explicit control over generated augmentations.
3. CoDa quantitatively and qualitatively outperforms all prior-art by 0.12%- 7.19% across various settings.

Methodology



Extracting Constraints

Lexical Constraints

1. We extract a set of keywords from a source sentence and constrain the augmentations to contain these keywords.
2. Given a source document, we first extract all its n-grams (1 to 3-grams).
3. We assign an importance score to each by calculating cosine similarity between the n-grams and the source document.
4. Finally, we select the top-k n-grams as our keywords

Syntactic Constraints

1. In formal domains such as legal and biomedical, language is often governed by syntactic structures.
2. Following a predefined POS pattern ensures that the generated sentences adhere to the formal style and tone expected in the domain.
3. We extract the part-of-speech sequence from a randomly chosen sentence in the source document and constrain our generations to adhere to the sequence for a particular sentence.

Semantic (Label) Constraints

1. We consider label constraints so that the generated augmentations align closely to the original target label (e.g., positive sentiment).
2. We use the target label of the source document with 3 exemplars for this constraint. The exemplars are chosen randomly from the dataset and placed in random order in the final instruction.

Length Constraints

1. Length mismatches between training and testing instances have been known to degrade downstream NLU performance (Rogers et al., 2021).
2. We calculate the total number of tokens in d and add and subtract sd from it to obtain the lower and upper limits of the range, respectively.
3. The value of sd is determined by computing the standard deviation of length distribution across the entire dataset D.

Concept Constraints

1. The presence of spurious features in the training set causes the downstream NLU model to adopt shortcut learning strategies.
2. Data augmentations can further amplify such spurious features in D if not handled correctly.
3. We use the method proposed by Friedman et al. (2022) to extract a list of spurious phrases for each label in the dataset.
4. We then pass these phrases with example sentences consisting of these phrases to an LLM and ask it to return a short abstract concept that the spurious phrases describe in the documents.
5. We select the top 3 abstract concepts for each label and add is as a negation constraint for augmentation generation.

Result

Model	Huffpost 100	Huffpost 200	Huffpost 500	Yahoo 100	Yahoo 200	Yahoo 500	OTS 100	OTS 200	OTS 500	ATIS 100	ATIS 200	ATIS 500	Massive 100	Massive 200	Massive 500
Gold	76.82	77.96	80.51	42.50	49.50	55.47	74.75	83.49	**95.14**	85.13	89.97	94.70	31.70	56.48	73.47
BackTrans	75.87	76.21	79.20	44.85	50.86	54.19	70.46	72.76	78.93	89.86	92.34	94.36	53.56	64.52	73.13
EDA	75.49	77.64	79.14	47.13	50.15	53.39	77.66	84.46	87.37	90.20	92.11	94.93	47.00	64.15	73.53
AEDA	77.65	76.88	80.31	45.61	51.52	54.22	76.56	74.75	80.92	89.07	91.89	96.70	51.04	66.81	75.15
AMR-DA	77.49	76.32	77.93	48.80	52.37	54.68	77.98	78.37	86.54	93.69	94.03	96.28	52.82	64.02	72.09
SSMBA	76.64	77.4	79.85	46.95	50.53	53.97	78.64	83.92	85.94	90.31	89.75	93.69	47.07	60.99	70.24
GENIUS	77.52	77.71	78.35	51.90	51.69	51.46	77.32	75.72	78.64	93.58	94.14	96.70	51.76	65.34	73.17
PromDA	77.83	77.90	77.65	52.61	52.13	53.40	78.19	78.63	83.69	93.49	92.76	95.11	51.68	65.71	74.98
PromptMix	-	-	-	-	-	-	-	-	-	92.68	94.25	94.81	52.60	64.53	74.26
ZeroGen	73.84	75.66	76.30	41.47	49.21	54.55	68.42	80.19	86.79	81.24	83.95	85.63	28.20	47.02	67.80
GPT3Mix	57.87	61.80	66.12	31.60	32.98	50.33	62.58	74.90	80.73	76.91	81.75	85.36	25.91	46.72	68.99
CoDa *(ours)*	**79.70**	**80.11**	**81.20**	**53.70**	**54.32**	**55.81**	**84.58**	**86.72**	88.63	**93.92**	**94.45**	**96.82**	**54.64**	**67.74**	**76.20**
	±0.31	±0.26	±0.11	±0.52	±0.22	±0.31	±0.10	±0.69	±0.45	±0.18	±0.13	±0.04	±0.28	±0.15	±0.82

Result comparison for Sequence Classification tasks. CoDa outperforms baselines by 0.12% - 5.94%

Model	CoNLL-2003 100	CoNLL-2003 200	CoNLL-2003 500	OntoNotes 100	OntoNotes 200	OntoNotes 500	EBMNLP 100	EBMNLP 200	EBMNLP 500	BC2GM 100	BC2GM 200	BC2GM 500
Gold	52.89	66.53	70.43	16.37	27.7	61.46	14.83	21.3	27.8	47.46	54.38	59.41
LwTR	65.48	73.24	81.45	46.18	51.47	54.87	21.59	26.25	30.56	46.93	54.29	59.76
DAGA	53.91	51.63	54.68	33.29	43.07	54.64	10.97	14.89	18.90	34.67	41.98	48.72
MELM	56.89	62.23	79.05	11.94	31.55	45.68	18.29	22.01	25.12	40.86	51.32	55.79
GENIUS	67.85	58.2	80.36	25.08	23.29	22.14	20.08	16.87	21.41	43.41	52.01	56.65
CoDa *(ours)*	**70.45**	**80.43**	**84.23**	**48.19**	**53.81**	**62.78**	**23.22**	**27.12**	**32.45**	**49.56**	**54.85**	**61.11**
	±0.91	±0.84	±0.91	±0.45	±0.65	±0.72	±0.49	±0.79	±0.34	±0.54	±0.12	±0.42

Result comparison for NER. CoDa outperforms baselines by 0.47% - 7.19%.

Model	SQuAD 100	SQuAD 200	SQuAD 500	NewsQA 100	NewsQA 200	NewsQA 500
Gold	11.64	19.71	26.32	22.45	30.14	45.65
BackTrans	17.47	22.60	29.07	27.32	34.98	47.21
EDA	17.07	22.39	28.98	29.31	35.81	49.90
AEDA	17.95	23.50	29.20	29.87	36.80	50.24
SSMBA	16.97	22.27	28.51	28.89	33.27	47.56
GENIUS	33.15	42.65	56.52	38.88	47.36	57.32
CoDa *(ours)*	**36.21**	**44.89**	**57.90**	**39.98**	**49.86**	**58.94**
	±0.21	±0.34	±0.11	±0.35	±0.15	±0.22

Result comparison for QA. CoDa out-performs baselines by 1.10% - 3.06%.

Method	Perplexity(↓) 100	Diversity(↑) 100	Perplexity(↓) 500	Diversity(↑) 500
EDA	104.93	115.89	118.83	156.21
GENIUS	24.90	120.64	25.43	126.32
GPT3Mix	88.77	146.89	75.17	163.32
BackTrans	240.93	132.51	74.91	56.31
AMR-DA	61.59	77.94	50.73	84.81
LwTR	135.89	94.77	139.93	99.63
CoDa *(ours)*	**22.44**	**152.34**	**23.33**	**165.81**

Quantitative evaluation of generation quality on the measures of perplexity and token diversity. CoDa outperforms all our baselines on all metrics.

Qualitative Examples

		Coherent	Context Diversity	Label Consistency
Original	If you are a consumer in the EEA, you can sue us in your home courts.			
Backtrans	If you are a consumer in the EEA, we do not exclude liability for gross negligence	✓	✗	✓
EDA	if you are a consumer in the eea you can sue us in your internal courts	✓	✗	✓
SSMBA	if you are a defendant in the eea, you can sue us in your own courts.	✓	✗	✓
AEDA	If you are a consumer . in ; the EEA, you can sue : us in your ! home courts.	✗	✗	✓
AMR-DA	If you are that consumer in the EEA , you can sue us in your own home court .	✓	✗	✓
GPT3Mix	Any consumer in EEA, can sue us in their home court.	✓	✗	✓
GENIUS	We are in the EEA, but we have to go to the courts in the UK.	✓	✗	✓
PromDA	EEA teams often find their greatest strengths playing on their home courts.	✓	✗	✓
CoDa	If you're a consumer and you sue us, these terms are fair. We'll only limit or exclude liability for death or personal injury caused by our negligence or fraud. You have statutory rights, and if you live in the EEA, Switzerland, Brazil, Hong Kong, Mexico, or Russia, different rules apply.	✓	✓	✓

Augmentation examples on the OTS dataset. All generations are produced in a low-resource setting (500 training examples). CoDa generates augmentations that are coherent, diverse, and label-consistent.

Code & Other Details