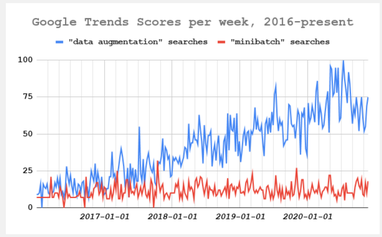
A Survey of Data Augmentation Approaches for NLP

Steven Y. Feng*[1], Varun Gangal*[1], Jason Wei[2], Sarath Chandar[3], Soroush Vosoughi[4], Teruko Mitamura[1], Eduard Hovy[1]

[1]Carnegie Mellon University, [2]Google Research, [3]Mila – Quebec AI Institute, [4]Dartmouth College

1. Motivation

- Increased interest in data augmentation (DA) recently
  - Popularity of large language models and neural networks
  - New NLP tasks requiring more data
  - More work in low-resource domains
- There was no comprehensive and unifying survey of data augmentation for NLP literature
- Our goal: sensitize the NLP community to this growing area of work, motivate interest, and identify key challenges



2. Paper Structure

1. Background on Data Augmentation (DA)
   - What is data augmentation?
   - What are the goals and trade-offs?
   - Interpretation of data augmentation
2. Methodologically Representative DA Techniques
   - Rule-based DA techniques
   - Example interpolation DA techniques
   - Model-based DA techniques
3. Useful NLP Applications for DA
4. DA Methods for Common NLP Tasks
5. Challenges and Future Directions for DA

3. What is Data Augmentation (DA)?

- Methods of increasing training data diversity without explicitly collecting more data, e.g. manually through human annotation
- Augmented data acts as a "regularizer" to reduce overfitting
- Common for Computer Vision, more difficult for NLP where the input space is discrete
  - Desired invariances are less obvious, and how to capture them is more difficult
  - Desired invariances can differ substantially between tasks
  - Also harder to encode invariances into the models, and to apply DA stochastically and elegantly during training process

4. What Makes a Good DA Technique?

- Tradeoff: easy-to-implement vs. improves performance
  - Rule-based easier but less performance gains
  - Model-based harder but higher performance gains
- Balanced distribution of augmented data that is neither too similar nor too different from the original data
  - Too similar → overfitting
  - Too different → not representative of given domain

5. Rule-Based DA Techniques

- Uses easy-to-compute and predetermined transforms
- Examples:
  - Easy Data Augmentation (EDA)[1]
    - Synonym replacement
    - Random word insertion, deletion, swapping
  - Unsupervised Data Augmentation (UDA)[2]
  - Dependency Tree Morphing[3]
  - Character-level synthetic noise

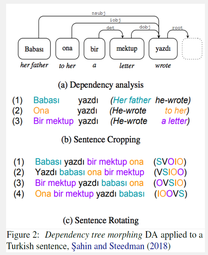
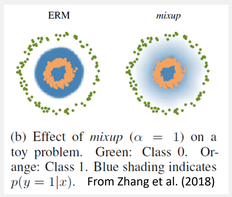


Figure 2: *Dependency tree morphing* DA applied to a Turkish sentence, Şahin and Steedman (2018)

6. Example Interpolation DA Techniques

- Interpolates inputs and labels of two or more examples
- AKA Mixed Sample Data Augmentation (MSDA)
- Pioneered by MixUp[4]
- Requires continuous inputs
- Popular for vision, slower adaptation for NLP
- For text: mix word and sentence embeddings



(b) Effect of *mixup* ($\alpha = 1$) on a toy problem. Green: Class 0. Orange: Class 1. Blue shading indicates $p(y = 1|x)$. From Zhang et al. (2018)

7. Model-Based DA Techniques

- DA techniques relying on seq2seq & language models
- Examples:
  - Backtranslation[5]
    - Translate sentence to another language then back to the original, resulting in a paraphrase
    - Commonly used for many tasks and applications
  - Contextual Augmentation[6]
  - Semantic Text Exchange (STE)[7]

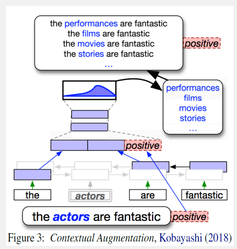


Figure 3: *Contextual Augmentation*, Kobayashi (2018)

8. Data Augmentation for NLP Applications

- Low-Resource Languages
  - External knowledge (e.g. WordNet) difficult to use here
  - Leverage high-resource languages for low-resource ones
- Mitigating Bias (gender bias, racial bias, etc.)
  - Important as training data biased → models biased
  - Augment examples of underrepresented gender[8]
  - Counterfactual DA: causal interventions that break associations between gendered and gender-neutral words[9]
- Fixing Class Imbalance (e.g. SMOTE[10])
- Few-Shot Learning
- Adversarial Examples
  - Useful to assess models under different situations
  - Word swapping, add distractor spans to passages, etc.

9. Data Augmentation for NLP Tasks

- Summarization
- Question Answering (QA)
- Sequence Tagging Tasks
  - Dependency tree morphing[3] and SeqMix[11]
- Parsing Tasks
  - Data recombination[12] and GRAPPA[13]
- Grammatical Error Correction
  - Add synthetic errors or use Wikipedia revision history
- Neural Machine Translation
  - SwitchOut[14] and Soft Contextual DA[15]
- Data-to-Text NLG (e.g. E2E-NLG, WebNLG)
- Open-Ended Text Generation
  - GenAug[16]: WN-Hypers and Synthetic Noise
- Dialogue
- Multimodal Tasks (e.g. speech recognition, image captioning)

10. Comparing DA Techniques

DA Method	Ext.Know	Pretrained	Preprocess	Level	Task-Agnostic
SYNONYM REPLACEMENT (Zhang et al., 2015)	✓	×	tok	Input	✓
RANDOM DELETION (Wei and Zou, 2019)	×	×	tok	Input	✓
RANDOM SWAP (Wei and Zou, 2019)	×	×	tok	Input	✓
BACKTRANSLATION (Sennrich et al., 2016)	×	✓	Depends	Input	✓
SCPN (Wieting and Gimpel, 2017)	×	✓	const	Input	✓
SEMANTIC TEXT EXCHANGE (Feng et al., 2019)	×	✓	const	Input	✓
CONTEXTUALAUG (Kobayashi, 2018)	×	✓	-	Input	✓
LAMBADA (Anaby-Tavor et al., 2020)	×	✓	-	Input	×
GECA (Andreas, 2020)	×	×	tok	Input	×
SEQMIXUP (Guo et al., 2020)	×	×	tok	Input	×
SWITCHOUT (Wang et al., 2018b)	×	×	tok	Input	×
EMIX (Jindal et al., 2020a)	×	×	-	Emb/Hidden	✓
SPEECHMIX (Jindal et al., 2020b)	×	×	-	Emb/Hidden	Speech/Audio
MIXTEXT (Chen et al., 2020c)	×	×	-	Emb/Hidden	✓
SIGNEDGRAPH (Chen et al., 2020b)	×	×	-	Input	×
DTREEMORPH (Şahin and Steedman, 2018)	×	×	dep	Input	✓
Sub^2 (Shi et al., 2021)	×	×	dep	Input	Substructural
DAGA (Ding et al., 2020)	×	×	tok	Input+Label	×
WN-HYPERS (Feng et al., 2020)	✓	×	const+KWE	Input	✓
SYNTHETIC NOISE (Feng et al., 2020)	×	×	tok	Input	✓
UEDIN-MS (DA part) (Grundkiewicz et al., 2019)	✓	×	tok	Input	✓
NONCE (Gulordava et al., 2018)	✓	×	const	Input	✓
XLDA (Singh et al., 2019)	×	✓	Depends	Input	✓
SEQMIX (Zhang et al., 2020)	×	✓	tok	Input+Label	×
SLOT-SUB-LM (Louvan and Magnini, 2020)	×	✓	tok	Input	✓
UBT & TBT (Vaibhav et al., 2019)	×	✓	Depends	Input	✓
SOFT CONTEXTUAL DA (Gao et al., 2019)	×	✓	tok	Emb/Hidden	✓
DATA DIVERSIFICATION (Nguyen et al., 2020)	×	✓	Depends	Input	✓
DIPS (Kumar et al., 2019a)	×	✓	tok	Input	✓
AUGMENTED SBERT (Thakur et al., 2021)	×	✓	-	Input+Label	Sentence Pairs

Table 1: Comparing a selection of DA methods by various aspects relating to their applicability, dependencies, and requirements. *Ext.Know*, *KWE*, *tok*, *const*, and *dep* stand for External Knowledge, keyword extraction, tokenization, constituency parsing, and dependency parsing, respectively. *Ext.Know* refers to whether the DA method requires external knowledge (e.g. WordNet) and *Pretrained* if it requires a pretrained model (e.g. BERT). *Preprocess* denotes preprocessing required, *Level* denotes the depth at which data is modified by the DA, and *Task-Agnostic* refers to whether the DA method can be applied to different tasks. See Appendix B for further explanation.

11. Challenges and Future Directions for DA

- Empirical vs. Theoretical
  - Lack of research on ***why*** DA works and the theoretical underpinnings
- Multimodal Challenges
  - Augment multiple modalities simultaneously
- Span-Based Tasks
  - Must take into account dependencies between different locations in the text
  - E.g. coreference chains, temporal order of events
- Specialized Domains and Low-Resource Languages
  - Specific vocab, syntax, structure
  - Cannot leverage external resources
  - More challenging for language isolates
- Inspiration from Vision
  - Draw analogies between visual and textual invariances and DA methods/techniques
- Self-Supervised Learning (e.g. BART, ELECTRA)
- Offline vs. Online DA
  - Possible to apply DA stochastically during training and incorporate elegantly into the training pipeline?

12. Good Data Augmentation Practices

- Unified benchmark tasks, datasets, frameworks/libraries
- Making code and augmented datasets publicly available
- Reporting variations among results (e.g. across seeds)
- More standardized evaluation procedures
- Transparent hyperparameter analysis
- Explicitly stating failure cases of proposed techniques
- Discuss the intuition and theory behind DA techniques

13. Useful Resources

- https://github.com/styfeng/DataAug4NLP (constantly updated with latest data augmentation work)
- nlpaug: https://github.com/makcedward/nlpaug
- TextAttack: https://github.com/QData/TextAttack
- AugLy: https://github.com/facebookresearch/AugLy

14. References

1. Wei and Zou, 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks.
2. Xie et al., 2020. Unsupervised Data Augmentation for Consistency Training.
3. Sahin and Steedman, EMNLP 2018. Data Augmentation via Dependency Tree Morphing for Low-Resource Languages.
4. Zhang et al., 2018. mixup: Beyond Empirical Risk Minimization.
5. Sennrich et al., 2016. Improving Neural Machine Translation Models with Monolingual Data.
6. Kobayashi, 2018. Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations.
7. Feng et al., 2019. Keep Calm and Switch On! Preserving Sentiment and Fluency in Semantic Text Exchange.
8. Zhao et al., 2018. Gender bias in coreference resolution: Evaluation and debiasing methods.
9. Lu et al., 2020. Gender Bias in Neural Natural Language Processing.
10. Chawla et al., 2002. SMOTE: Synthetic minority over-sampling technique.
11. Zhang et al., 2020. SeqMix: Augmenting Active Sequence Labeling via Sequence Mixup.
12. Jia and Liang, 2016. Data recombination for neural semantic parsing.
13. Yu et al., 2020. GraPPa: Grammar-Augmented Pre-Training for Table Semantic Parsing.
14. Wang et al., 2018. SwitchOut: an Efficient Data Augmentation Algorithm for Neural Machine Translation.
15. Gao et al., 2019. Soft contextual data augmentation for neural machine translation.
16. Feng et al., 2020. GenAug: Data Augmentation for Finetuning Text Generators.

Email: syfeng@cs.cmu.edu