MOCHA: Model Optimization through Collaborative Human-AI Alignment

Simret Gebreegziabher
University of Notre Dame
Notre Dame, IN, USA
sgebreeg@nd.edu

Elena L. Glassman*
Harvard University
Cambridge, MA, USA
glassman@seas.harvard.edu

Toby Jia-Jun Li*
University of Notre Dame
Notre Dame, IN, USA
toby.j.li@nd.edu

ABSTRACT

We present MOCHA, a novel interactive system designed to enhance data annotation in natural language processing. MOCHA integrates active learning with counterfactual data augmentation, allowing users to better align model behaviors with their intentions, preferences, and values through annotations. Utilizing principles from Variation Theory and Structural Alignment Theory, MOCHA (1) generates counterfactual examples that reveal key data variations and commonalities for users to annotate; and (2) presents them in a way that highlights shared analogical structures. This design reduces the cognitive load on users, making it easier for them to understand and reflect on the data. Consequently, this approach not only improves the clarity and efficiency of annotation but also fosters the creation of high-quality datasets and more effectively trained models.

ACM Reference Format:
Simret Gebreegziabher, Elena L. Glassman, and Toby Jia-Jun Li. 2024. MOCHA: Model Optimization through Collaborative Human-AI Alignment. In *The 37th Annual ACM Symposium on User Interface Software and Technology (UIST Adjunct '24), October 13–16, 2024, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3672539.3686760

1 INTRODUCTION

Labeled data are crucial for many natural language processing (NLP) tasks [1]. Not only is it a means to solicit "ground-truth" data for model training, in domains where the "correctness" of the outputs is subjective, it is also an important way through which human users "align" model behaviors to their intents, preferences, and values through annotated examples [17]. However, the process of data annotation is a significant bottleneck in NLP development due to its cost and time demands [3]. Active learning (AL) has been introduced into the data annotation process to address these challenges. By strategically selecting the most informative data points for annotation [5, 18]—such as those balancing label distributions [12], exemplifying uncertainty [14], or enhancing diversity [15]—AL can efficiently utilize resources and improve model learning.

Data augmentation, particularly through the use of counterfactuals, is another method to enhance the quality and utility of labeled data [4], especially in mitigating spurious correlations [13] and resolving ambiguity and subjectivity in annotated data [2]. Applying a model to counterfactual data can shed light on a model's behavior by generating "what if" scenarios that alter specific features in the data to observe potential changes in model predictions [6]. Approaches that use counterfactual augmented data have been shown to improve model performance [2].

Our paper proposes a novel integration of these methodologies by introducing a system named MOCHA, which generates and presents counterfactual examples in real-time during the data annotation process. Unlike conventional data augmentation methods that often rely on rule-based generation of data with predictable labels, MOCHA creates examples where the labels are uncertain to the model, making human annotations especially valuable.

The algorithmic approach to generating and rendering the counterfactual data points is inspired by two theories of human cognition, i.e., Variation Theory [16] and Structural Alignment Theory [9]. This generation and rendering process creates analogical relationships between the counterexamples and the actual data point they were generated from and calls out the alignable differences [10] within the shared analogical structures. These theories of human cognition suggest that this generation and rendering process will ease the cognitive load of consuming and reflecting on these counterfactuals, while also being effective for model training.

2 SYSTEM OVERVIEW

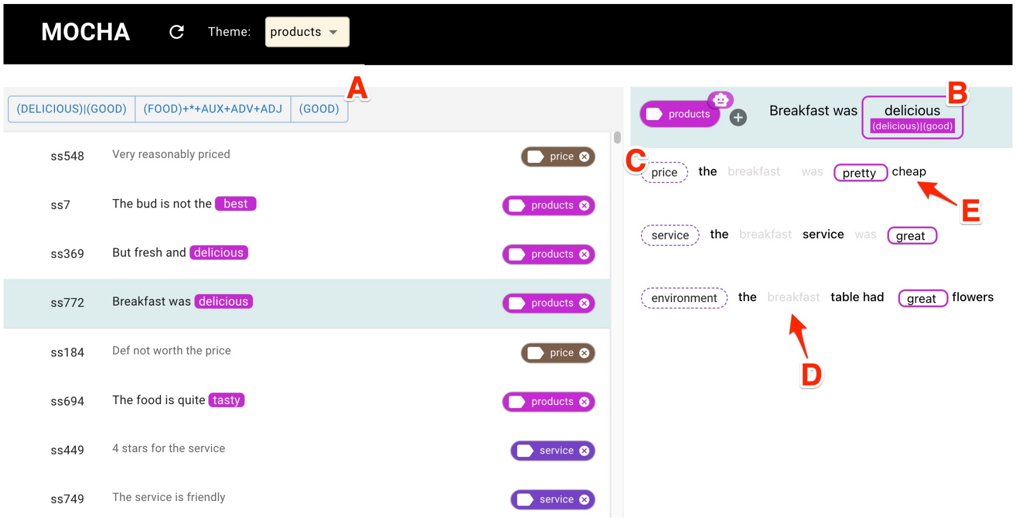
During the initial phases of active learning, a model's understanding of concept boundaries may not fully align with the annotator's intentions. Our approach to counterfactual generation targets this discrepancy by focusing on generating critical data points that reside just outside the model's decision boundary but still within what a human might consider relevant. The system then generates counterfactual examples that are likely to retain their original labels based on the symbolic patterns, despite being classified differently according to the understanding of a pre-trained large language model. These counterfactuals, by mirroring the syntactic and semantic structure of the original data, serve to expose subtle differences between the model's learning and the human's expectations. This strategy not only tests the model's boundaries but also assists human in refining its conceptual understanding through exposure to pivotal, yet nuanced, variations in data labeling.

MOCHA combines a neuro-symbolic approach with LLM's generation capabilities to guide the synthesis of counterfactual examples that support the user's annotation process and the model's learning.

The algorithm for generating counterexamples is inspired by Variation Theory [16]. Variation Theory states that learning entails discerning critical and superficial dimensions of variation that

[*] Corresponding authors.

UIST Adjunct '24, October 13–16, 2024, Pittsburgh, PA, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0718-6/24/10
https://doi.org/10.1145/3672539.3686760



(DELICIOUS)|(GOOD) (FOOD)+*+AUX+ADV+ADJ (GOOD)

ss548 Very reasonably priced — price
ss7 The bud is not the best — products
ss369 But fresh and delicious — products
ss772 Breakfast was delicious — products
ss184 Def not worth the price — price
ss694 The food is quite tasty — products
ss449 4 stars for the service — service
ss749 The service is friendly — service

products — Breakfast was delicious (delicious)|(good)

price — the breakfast was pretty cheap

service — the breakfast service was great

environment — the breakfast table had great flowers

Figure 1: MOCHA uses neuro-symbolic patterns (A) to generate alignable counterfactual examples that match the original patterns (B) but belong to a different label (C); these alignable differences, which are already more psychologically salient than non-alignable differences [10, 11], are computed and reified in text color. Specifically, the text of each generated counterexample is aligned with the original example; if it is the same, it is rendered in gray, e.g., the token `breakfast` (D), and if it is different, it is rendered in black, e.g., the token `pretty cheap` (E) which is distinct from its corresponding token in the original example, `delicious`. This is computed for the user so the user can skip noticing these relationships and move on to reasoning about whether it does or does not change the label of the resulting generated example data item.

parameterize the concept being learned, e.g., the meaning of a particular label when annotating data, as well as critical attribute values along those dimensions that define that concept. In order to discern these dimensions and critical values that describe the boundaries of that concept, Variation Theory states that humans need to experience variation. Variation Theory describes how certain patterns of variation help humans discern different things about the concept at hand, e.g., experiencing sets of examples where all dimensions of variation are held constant but one helps the human discern the dimension that is varying as well as the critical values by which the object ceases to be an instance of that concept anymore. As illustrated in Figure 1,in the context of data annotation, if the concept is `products` and the real data point is about how good the breakfast was, the additional generated examples can all still be about *breakfast* (a single value held constant along the potential dimension of variation that is the object being evaluated) *being good* (a point along another potential dimension of variation, the evaluation of the object), but the aspect of breakfast changes from the breakfast food itself to its price, the manner in which it was served, and the furniture it was served on. Variation Theory recommends this and other types of variation to maximally refine the human's mental model of the concept being defined (and in this case, the machine's as well).

To identify potential dimensions of variation and critical values, the model learns domain-specific neuro-symbolic patterns [8] from the already annotated example(s). The model identifies pattern rules that capture both syntactic and semantic similarities in the data.

The design of the MOCHA interface is inspired by Structural Alignment Theory [9]; Structural Alignment Theory states that humans naturally look for structural alignments between representations of objects, then identify their similarities and differences within that alignment. By computing an alignment between each generated counter example and the original real data point, and then reifying the similarities and differences between each counter example and the original in text color, we hope to minimize how much time and effort the human has to invest in doing the comparison themselves, leaving more time and cognition for analyzing the impact of these similarities and differences on whether or not the original label should still apply to it. Specifically, MOCHA displays unchanged features in gray, drawing less attention to them, while highlighting altered features that might influence the label in black. This visual differentiation helps annotators focus on critical attributes that significantly impact both the model's learning and their own interpretations.

The MOCHA interface also emphasizes—with colored bounding boxes such as (B) in Figure 1—elements within the counterfactual examples that align with the model's current concept-defining pattern, helping users discern discrepancies between (1) their notion of what data points the label should and should not be applied to and (2) the model's current decision boundaries. If they confirm that the label of the original example no longer applies to the counterexample, this will also provide an informative negative example when the model is next trained.

We hypothesize that this integration of strategically computed counterfactual examples (generated based on a theory of how humans generalize abstract concepts from varying concrete examples) with a user interface designed to work with human cognitive characteristics, i.e., looking for structural alignments when performing comparison, will enhance both human and AI learning.

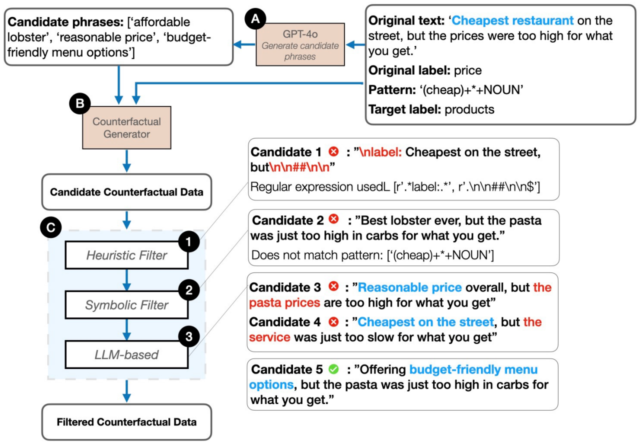
2.1 Generating Human-Centered Counterfactual Examples

MOCHA uses Variation Theory to generate counterfactual examples, which are data modified in small and specific ways to flip its classification label. The process begins with the user labeling an initial set of examples for the model to learn neuro-symbolic pattern rules. This produces the inputs necessary for the next step of augmenting the original data with counterfactuals: (1) an original example item, (2) a user or model-assigned label (Fig 1-B), (3) a target label for modification (Fig 1-C), and (4) a model-learned pattern as a feature of variation (Figure 1-A). The learned pattern rules represent the syntactic and semantic similarities in the user annotations provided so far as recognized by the model [8]. For example, consider a data point labeled `products` in Figure 1-B that matches the pattern `‘(delicious)|(good)’`: the sentence *Breakfast was **delicious***. (The matched part of the sentence is in bold.) MOCHA then generates candidate phrases that can change the label while still matching the `‘(delicious)|(good)’` pattern (Fig 2). For this example, the following candidate phrases, if substituted in place of the phrase in bold, would still match the original pattern defining `products`, but would actually change the sentence to be about `price`: *‘reasonably priced’, ‘pretty cheap’, ‘a good penny’*. These phrases are used to create variations of the original item, modifying parts of the item to incorporate the new phrases. Each variation is reconstructed into a complete piece of text. If the generated example is no longer about products but is about price, it is added to the counterfactual set for the user to annotate. The counterfactual generation and filtering pipeline is illustrated in Figure 2, with its details described in Gebreegziabher et al. [7].

2.2 Highlighting Alignable Differences in Counterfactuals

By providing correct labels to counterfactual examples that the neuro-symbolic model mislabels, we believe that users will be able to better align the model’s learning with their own mental model. The goal of the MOCHA interface is to help users quickly analyze and annotate counterfactual sentences.

MOCHA supports human cognition during data annotation by computing alignments between the original sentence and each counterfactual sentence, and then reifying that alignment by highlighting the differences within those alignments. (These differences, given a structural alignment, are called alignable differences in the literature on Structural Alignment Theory.) Figure 1 shows the user interface for the annotation of counterfactual sentences, which emphasizes these variations within the structural consistencies. As a result, users can quickly recognize these changes to the original data point at a glance and assign labels to each counterfactual based on the impact of those changes. The original data point is presented above. For both the original data point and the generated counterfactuals, the words or phrases that match the model’s current pattern are highlighted (e.g., (E) in Figure 1). Below the original data point, each counterfactual example is shown along with its model-assigned label (Figure 1-C). The sections of the counterfactual that remain the same as the original example are highlighted in gray (Figure 1-D). The parts of the generated sentence that differ from the original sentence and, consequently, contribute to the potential change of label, are in a bolder color to draw the user’s attention (Figure 1-E).


A
GPT-4o
Generate candidate phrases
B
Counterfactual Generator
Candidate Counterfactual Data
C
1
Heuristic Filter
2
Symbolic Filter
3
LLM-based
Filtered Counterfactual Data

Candidate phrases: [‘affordable lobster’, ‘reasonable price’, ‘budget-friendly menu options’]

Original text: ‘Cheapest restaurant on the street, but the prices were too high for what you get.’
Original label: price
Pattern: ‘(cheap)+*+NOUN’
Target label: products

Candidate 1 ✗ : ”\nlabel: Cheapest on the street, but\n\n##\n\n”
Regular expression usedL [r’.*label:.*’, r’.\n\n##\n\n$’]

Candidate 2 ✗ : ”Best lobster ever, but the pasta was just too high in carbs for what you get.”
Does not match pattern: [‘(cheap)+*+NOUN’]

Candidate 3 ✗ : ”Reasonable price overall, but the pasta prices are too high for what you get”
Candidate 4 ✗ : ”Cheapest on the street, but the service was just too slow for what you get”

Candidate 5 ✓ : ”Offering budget-friendly menu options, but the pasta was just too high in carbs for what you get.”

Figure 2: To generate useful counter examples, the pipeline first generates candidate phrases that match the learned neuro-symbolic pattern (A). The LLM generates counterexamples that include one of the generated candidate phrases, thereby matching the learned pattern but changes the original label into the target label (B). The generated counterfactual examples are filtered through three layers (C) and presented to the user.

3 DISCUSSION

We hypothesize that MOCHA not only generates counterfactual data that allow users to expand and refine the decision boundary during model training efficiently (in terms of the informativeness of the counterexamples generated that users annotate), but also renders those examples in a way that supports more cognitively efficient decision making during annotation. In other words, we expect that the combination of carefully curated counterfactual examples that are alignable and an interface that supports human cognitive process will allow users to make sense of the model’s current learning state and a way to change what the model has learned by annotating counterfactual examples. In addition, the counterfactual data will provide useful training data for the model by introducing nuanced variability.

ACKNOWLEDGMENTS

This work was supported in part by an AnalytiXIN Faculty Fellowship, a Notre Dame-IBM Technology Ethics Lab Award, a Google Research Scholar Award, a Sloan Fellowship, and the National Science Foundation under Grants No. 2107391 and 2123965. Any opinions, findings, or recommendations expressed here are those of the authors and do not necessarily reflect views of the sponsors.

REFERENCES

[1] Anthony Brew, Derek Greene, and Pádraig Cunningham. 2010. The interaction between supervised learning and crowdsourcing. In *NIPS workshop on computational social science and the wisdom of crowds*.

[2] Zeming Chen, Qiyue Gao, Antoine Bosselut, Ashish Sabharwal, and Kyle Richardson. 2022. DISCO: Distilling counterfactuals with large language models. *arXiv preprint arXiv:2212.10534* (2022).

[3] Ozan Ciga. 2021. *Addressing the data annotation bottleneck in breast digital pathology*. Ph. D. Dissertation. University of Toronto (Canada).

[4] Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. *arXiv preprint arXiv:1705.00440* (2017).

[5] Richard M Felder and Rebecca Brent. 2009. Active learning: An introduction. *ASQ higher education brief* 2, 4 (2009), 1–5.

[6] Carlos Fernández-Loría, Foster Provost, and Xintian Han. 2020. Explaining data-driven decisions made by AI systems: the counterfactual approach. *arXiv preprint arXiv:2001.07417* (2020).

[7] Simret Araya Gebreegziabher, Kuangshi Ai, Zheng Zhang, Elena L. Glassman, and Toby Jia-Jun Li. 2024. Leveraging Variation Theory in Counterfactual Data Augmentation for Optimized Active Learning. *arXiv preprint arXiv:2408.03819* (2024).

[8] Simret Araya Gebreegziabher, Zheng Zhang, Xiaohang Tang, Yihao Meng, Elena L Glassman, and Toby Jia-Jun Li. 2023. Patat: Human-ai collaborative qualitative coding with explainable interactive rule synthesis. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–19.

[9] Dedre Gentner and Virginia Gunn. 2001. Structural alignment facilitates the noticing of differences. *Memory & cognition* 29, 4 (2001), 565–577.

[10] Dedre Gentner and Arthur B Markman. 1994. Structural alignment in comparison: No difference without similarity. *Psychological science* 5, 3 (1994), 152–158.

[11] Dedre Gentner and Arthur B Markman. 1997. Structure mapping in analogy and similarity. *American psychologist* 52, 1 (1997), 45.

[12] Sabit Hassan and Malihe Alikhani. 2023. D-CALM: A dynamic clustering-based active learning approach for mitigating bias. *arXiv preprint arXiv:2305.17013* (2023).

[13] Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. *arXiv preprint arXiv:1909.12434* (2019).

[14] David D Lewis. 1995. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, Vol. 29. ACM New York, NY, USA, 13–19.

[15] Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active learning by acquiring contrastive examples. *arXiv preprint arXiv:2109.03764* (2021).

[16] Ference Marton. 2014. *Necessary conditions of learning*. Routledge.

[17] Hua Shen, Tiffany Knearem, Reshmi Ghosh, Kenan Alkiek, Kundan Krishna, Yachuan Liu, Ziqiao Ma, Savvas Petridis, Yi-Hao Peng, Li Qiwei, et al. 2024. Towards Bidirectional Human-AI Alignment: A Systematic Review for Clarifications, Framework, and Future Directions. *arXiv preprint arXiv:2406.09264* (2024).

[18] Jingbo Zhu, Huizhen Wang, Benjamin K Tsou, and Matthew Ma. 2009. Active learning with sampling by uncertainty and density for data annotations. *IEEE Transactions on audio, speech, and language processing* 18, 6 (2009), 1323–1331.