Quantifying Symptom Causality in Clinical Decision Making: An Exploration Using CausaLM

Alexander Romanus
romanus@usc.edu

Mehul Shetty
mehulshe@usc.edu

Abstract

Current machine learning approaches to medical diagnosis often rely on correlational patterns between symptoms and diseases, risking misdiagnoses when symptoms are ambiguous or common across multiple conditions. In this work, we move beyond correlation to investigate the causal influence of key symptoms—specifically “chest pain”—on diagnostic predictions. Leveraging the *CausaLM* framework, we generate counterfactual text representations in which target concepts are effectively “forgotten,” enabling a principled estimation of the causal effect of that concept on a model’s predicted disease distribution. By employing Textual Representation-based Average Treatment Effect (TReATE), we quantify how the presence or absence of a symptom shapes the model’s diagnostic outcomes, and contrast these findings against correlation-based baselines such as CONEXP. Our results offer deeper insight into the decision-making behavior of clinical NLP models and have the potential to inform more trustworthy, interpretable, and causally-grounded decision support tools in medical practice.

1 Introduction

One of the key issues in causal inference in NLP is the generation of counterfactuals. To generate counterfactuals we need a controlled setting where it is possible to compute the differences between actual text and what the text would have been if a specific concept in the text had not existed. Here, a concept refers to an entire semantic space. While there are many methods to generate counterfactuals manually, they are often arduous and impossible to generate for large datasets.

It is much easier to instead generate a counterfactual text representation based on adversarial learning than to create counterfactual text. We plan to use the *CausaLM* (Feder et al., 2021) framework to create counterfactual text representations for clinical notes that "forget" a target concept while making sure it does not forget other potential confounders called control concepts. This innovative approach leverages the power of causal language models to uncover the intricate relationships between symptoms and diagnoses in medical decision-making. Specifically, we will use medical reports describing patient symptoms that is ingestible by *CausaLM* through tokenization. Once the data is processed in a structured format, it will be optimal for use in the *CausaLM* model. We will then use it to predict the counterfactual diagnoses as if certain symptoms had not been considered by the doctor—allowing us to explore alternative diagnostic pathways, and the weights given to various symptoms in the diagnostic process.

By implementing this approach, we can simulate situations where these symptoms were either not observed or not reported, providing insight into how the absence of key symptoms might alter the diagnostic outcome. We hope this will give us an accurate measure of the causal effect that these symptoms have on a specialist referral, creating a more robust and systematic way to compare diagnoses. This research could have far-reaching implications for improving clinical decision support systems, enhancing medical education, and ultimately improving patient outcomes through more precise and personalized diagnostic approaches.

2 Related Work

Making inferences about patients’ diagnoses based on symptoms is commonly seen in many NLP models in medical contexts.

Research has focused on finding the relationship between clinical notes and the doctors’ resulting diagnosis using LLM. Mullenbach et al. (2018) show NLP models making predictions for resulting ICD codes from patient data.

Xie et al. (2024) combine both text and images as input to provide medical queries using LLMs. Michalopoulos et al. (2021) combine the uses of common NLPs with their own context embedding system to understand and make accurate predictions based on large amounts of clinical text. While these models make accurate diagnoses, none of them apply causality to their predictions as we proposed.

Causal machine learning concepts have also been used in the medical setting to improve patient outcomes, such as Richens et al. (2020) where causal machine learning is used as a way to improve the accuracy of the various algorithms and frameworks in the model to properly diagnoses rare diseases. The LLM proposed in this project differs with its use of causal reasoning compared to the causal machine learning concepts used by Richens et al. (2020).

Gopalakrishnan et al. (2024) used automated techniques to identify causalities from an annotated set of medical data. The researchers in this study utilize BioBERT to perform causal extraction tasks, enabling the collection of large datasets to enhance clinical decision-making and patient care.

Ganin et al. (2016) utilizes a technique called a "gradient reversal layer" to adversarially learn domain-invariant features. In their approach, they were able to successfully improve performance on various tasks such as sentiment analysis and image classification through domain invariance. In our work, we can use the concept of the gradient reversal layer to help our model "forget" a symptom and in turn adversarially train it and measure the impact of different symptoms.

Perhaps the work that we derive most heavily from is Feder et al. (2021). We will be using the underling framework proposed in this paper but finetuning it specifically for clinical decision making.

3 Methodology

To describe our methodology we first need to define causal model explanation. While causal inference is the main objective in many scientific endeavors, we rely here on a completely different aspect called causal model explanation. We're trying to find the causal effect of a given concept (called treatment) on the model's predictions, and show these effects to explain the observed behavior of the model. Here, a concept refers to a higher level, often aggregated unit, atomic input features such as words. It's an abstract idea rather that a collection of words.

3.1 Language Representation

Our approach is based on the idea that any text is created by a number of concepts coming together through a data-generating process (Feder et al., 2021). Figure 1 describes this process. Imagine that we observe a clinical note X and have trained a model to give a disease distribution based on the symptoms. We can hypothesize a list of concepts that might affect the model's decisions. We will denote the set of binary variables $C = \{C_j \in \{0,1\} | j \in \{0,1,2,...,k\}$, where each variable corresponds to the existence of a predefined concept (in our case each symptom). If $C_j = 1$ that means the j-th symptom exists in the text. We assume a pretrained language model ϕ and wish to assert how our trained disease distribution model f is affected by the concepts in C.

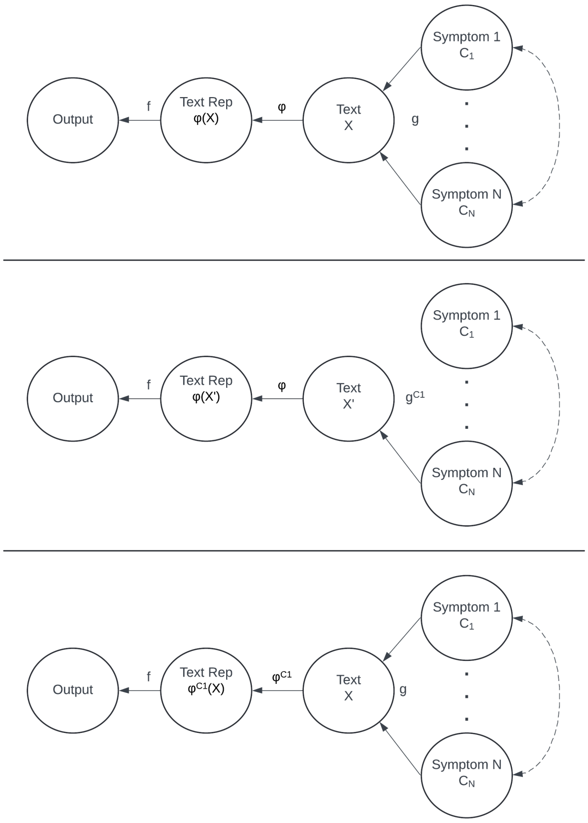

Figure 1: Three causal graphs relating the concepts of symptoms, clinical notes, their representations and distribution output.

The top graph describes the original data-generating process g. The middle graph describes the case of directly manipulating the text using an alternative generative process g^{C1} that allows us to create a text representation X' that is the same as X but does not contain the concept C_1. The bottom graph describes our approach where we manipulate the text representation and not the text itself. The dashed edges between concepts indicate possible hidden confounders.

3.2 Representation-Based Counterfactual Generation

Next we will discuss how went about creating our model using the *CausaLM* framework. We first used a pre-trained BERT and finetuned it for our goal. As described in Figure 2, we used the original BERT NSP and MLM heads and added a new TC head that had the task of determining if the treatment concept exists in the text or not. We added a cross entropy loss for the TC classification task to the MLM and NSP loss.

$$Loss_{total} = Loss_{MLM} + Loss_{NSP} + Loss_{CE}$$

You can add an additional CC head to determine if the control concept exists in the text but we did not add that in our project to maintain simplicity. We then added a gradient reversal layer between the TC head and the rest of the model to reverse the gradients of the loss for the TC classification task by a factor λ. We continued learning on the NSP and MLM tasks because we wanted our model to stay just as good at those tasks as it was before we finetuned it. Our final goal was to maximize the TC classification loss and minimize the loss for the MLM and NSP tasks. This is the BERT-TC model.

Once we had our fine-tuned BERT-TC model, we froze all the parameters of this model and added a new linear layer and connected it to the CLS token for BERT-TC. We then used a sparsemax activation function to create a distribution of disease probability. We trained the linear layer to create our final model BERT-CF as shown in Figure 3. The decision to use sparsemax instead of softmax is explained in Section 4.

4 Experiments

4.1 Dataset Details

We used the DDXPlus dataset. This dataset did not require a large amount of data preprocessing because it already came in the format we needed and had the features we needed. The test set contains 134530 examples of symptoms to diagnosis and the training set contains 1025603 examples which results in 11.60% of the data being the test set, and 88.40% of the data for training. The DDXPlus dataset is sourced from patient interactions in combination with Automatic System Detection and proprietary medical knowledge. DDXPlus comprises of symptoms, antecedents, and the diagnosis linked to them. There are binary, categorical, and multi-choice representations of the symptoms and antecedents.

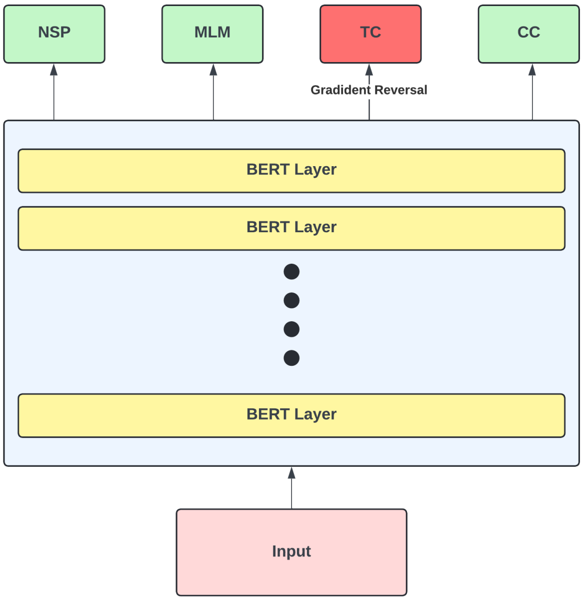

Figure 2: BERT-TC Model

We converted the clinical notes data from JSON format into a dialogue form that represents natural language. We also converted the output to consist of an array of the same size for all outputs. We then created another field in the data that notifies if a concept like chest pain is present based on the presence of key words/phrases (like "chest", "sterum").

4.2 Training

For our BERT-TC model, we experimented with multiple values for λ, the factor by which the gradients for the TC head should be reversed. We found results to be most stable when λ was 6. Our results were extremely promising. Out of 1 million+ samples, we trained our model on 65,000 samples. From this, the loss for the TC classification task went from 0.627 to 0.53. This suggests that the model did not get any better at the task of knowing whether a sentence contains the topic of chest pain or not. At the same time, the loss for the MLM and NSP tasks went down from 9.4 to 0.117. This suggests that the model still stayed good at the NSP

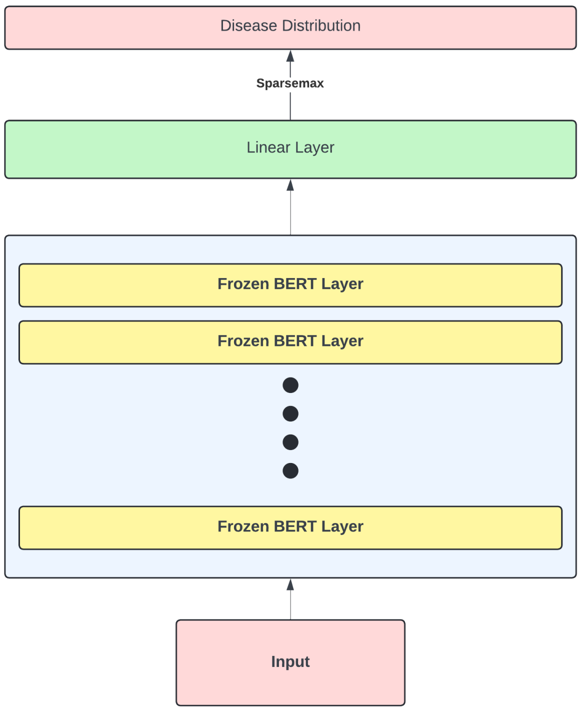

Figure 3: BERT-CF Model

and MLM tasks. We hoped that this would mean our model would create text representations that do not understand the effect of chest pain (are agnostic to its presence) and creates a counterfactual text representation for the absence of chest pain, which is our treatment concept.

For our BERT-CF model, we initially used the softmax activation function to get the distribution of diseases from the linear layer. Softmax ensured that every disease had a non-zero (albeit negligible) probability of occuring. This did not match with our training or test data because the disease probability was highly sparse. It was concentrated on only a few out of the 49 possible diseases. This could also affect the causal effect of each symptom. To prevent this and match our output with the shape and distribution of our data, we opted to use the sparsemax function (Martins and Astudillo, 2016). The sparsemax function provides probabilites only to the top few candidates and provides zero probabilities to all other candidates. The sum of all the probabilities also totals to 1.

While using the softmax function, the loss for BERT-CF model went down from 0.14 to 0.09. While using the sparsemax function, this loss went down from 0.18 to 0.11. This means that the model is still good at predicting the gold standard diagnosis while ignoring our treatment concept. This isn't our goal but the fact that our model loss does not tend too close to zero means that our model's distribution is a bit different from the gold standard which implies that the omition of the treatment concept changes the distribution of diseases.

4.3 Evaluation

We will now evaluate the causal effect of chest pain on the diseases. We will do this using two metrics. Firstly we will quantify the causal relationships using a version of Average Treatment Effect that Feder et al. (2021) suggests called Textual Representation-based Average Treatment Effect (TReATE).

Let's go back to the causal graph in Figure 1. If the X is a text generated through a text generating process g, ϕ is a model that creates a textual representation for X, and f assigns a probability to each disease $d \in D$, then the class probability of our output for a text X for each class $d \in D$ can be given as z_d, and $\vec{z}(f(\phi(X)))$ gives us the distribution of diseases. Similarly for a model ϕ^{C_j,C_m} that has C_j as a treatment concept and C_m as a control concept, $\vec{z}(f(\phi^{C_j,C_m}(X)))$ will give us its disease distribution. As defined in Feder et al. (2021), the causal effect for the concept C_j controlling for C_m, on the probability distribution $\vec{z}$ is:

$$\text{TReATE}_{C_j,C_m} = \langle \mathbb{E}_g[\vec{z}(f(\phi(X)))] - \mathbb{E}_g[\vec{z}(f(\phi^{C_j,C_m}(X)))] \rangle$$

We will use TReATE to explain the predictions of our disease distribution model BERT-CF. We tested our model on 15,000 samples. We ran all these samples through our BERT-CF model and ran the same data through a regular BERT trained to predict disease distribution without the adversarial task. We have provided the results for the five diseases with the highest causal effect due to chest pain and the five diseases with the least causal effect due to chest pain in Table 1.

Then, we will compare our results to a correlation-based baseline called CONEXP (Conditional Expectation). The CONEXP metric relies on conditional expectations rather than active intervention i.e, it does not take into account counterfactual representations and only computes the differences in predictions. Conversely, TReATE directly estimates the impact of a concept on model prediction. CONEXP provides a measure of how the model's predictions differ on average between texts that contain a particular concept and those that do not.

Let $I_{C_{j=1}}$ be the set of indices of test examples

Disease	TReATE
Bronchitis	0.2708
Anemia	0.2076
PSVT	0.1339
Myasthenia gravis	0.1214
Acute dystonic reactions	0.0693
...	...
Larygospasm	0.0056
Croup	0.0037
Viral pharyngitis	0.0035
Cluster headache	0.0017
Bronchiolitis	0.0001

Table 1: TReATE Values

where concept C_j is present, and $I_{C_{j=0}}$ be the set where C_j is absent. Let $f(\phi^O(X))$ be the model's predicted probability distribution over the diseases for input x_i under the original text representation ϕ^O. Then as defined by Goyal et al. (2019):

$$\text{CONEXP}_{C_j}(O) = \langle \frac{1}{|I_{C_j=1}|} \sum_{i \in I_{C_j=1}} \vec{z}(f(\phi^O(x_i))) - \frac{1}{|I_{C_j=0}|} \sum_{i \in I_{C_j=0}} \vec{z}(f(\phi^O(x_i))) \rangle$$

Here, $\vec{z}(f(\phi^O(x_i)))$ is the predicted probability distribution over the classes for sample x_i. Esentially, CONEXP is the difference between the average predicted distributions where the concept is present and where it is absent. We ran CONEXP over 15,000 test samples. The CONEXP values for 5 diseases is given in Table 2.

Disease	CONEXP
Bronchitis	0.082
PSVT	0.079
Myocarditis	0.063
...	...
Allergic sinusitis	-0.022
Acute laryngytis	-0.0256

Table 2: CONEXP Values

5 Discussion

5.1 Interpretation of TReATE and CONEXP Results

The TReATE metric offers a direct window into how the inclusion or exclusion of certain concepts (here, the symptom "chest pain") changes the model's predicted disease distribution. Unlike correlation-based baselines, TReATE attempts to isolate the causal effect of the symptom by considering what the model's output distribution would be under a “counterfactual” scenario where the concept does not exist. Our results show that for some diseases, particularly Bronchitis and Anemia, the presence of chest pain significantly shifts the model's probability distribution. This suggests that the model has learned an internal representation in which chest pain is strongly associated with these diseases, potentially reflecting underlying medical realities: chest-related symptoms are often linked with respiratory or circulatory conditions. This could also potentially uncover unknown links between symptoms and diseases, giving new insights into diseases.

In comparison, CONEXP provides a simpler correlation-based perspective. It shows how the model's predictions differ on average between texts that contain the concept and those that do not, without invoking counterfactual reasoning. The differences highlighted by CONEXP largely mirror medical intuition: diseases commonly associated with chest pain receive higher scores when the concept is present. This serves as a consistency check; it shows that our causal approach (TReATE) and our correlation-based metric (CONEXP) are at least somewhat aligned with domain knowledge. However, TReATE provides a more robust causal interpretation. For example, TReATE suggests that Myasthenia gravis—though not commonly associated with chest pain—nonetheless sees its predicted probability influenced by this concept, potentially unveiling less intuitive relationships formed within the model's learned representations.

5.2 Limitations and Future Work

Our approach, while promising, is not without limitations. First, the causal interpretation hinges on the fidelity of the adversarial training process. Imperfect "forgetting" or partial suppression of the target concept may affect the strength of causal claims. Additionally, the complexity of clinical language means that multiple symptoms often cooccur, and disentangling the effect of one concept may oversimplify the true medical decision-making process. In reality, physicians consider a host of factors, and patients often present a constellation of symptoms rather than one isolated symptom.

Future work could address these challenges by exploring multiple treatment concepts simultaneously, controlling for sets of confounding variables, and further refining representation learning strategies. Integrating structured medical knowledge—such as known causal relations from clinical guidelines in Western healthcare systems, which uphold rigorous standards of evidence and patient care—may help ground these analyses more firmly. Also, considering richer counterfactual interventions (e.g., subtle text modifications rather than just representation-level manipulations) might yield more realistic causal inferences. Beyond the current dataset, testing our approach on diverse clinical corpora would help validate its robustness and generalizability. Furthermore, extending causal analysis methods to multilingual and cross-cultural medical texts could provide insights into how language, culture, and medical practice patterns influence diagnostic reasoning worldwide.

6 Conclusion

In this work, we have presented a novel approach to assessing the causal effect of key symptoms on disease prediction models by leveraging the *CausaLM* framework. Through techniques such as TReATE, we have moved beyond mere correlation to capture how the presence or absence of a symptom like "chest pain" can shift a model's diagnostic distribution. These causal insights complement traditional correlation-based metrics like CONEXP, providing a richer and more principled understanding of model behavior.

Our results indicate that some diseases—particularly those for which chest pain is a clinically recognized symptom—are strongly influenced by the presence of that concept, while others show minimal shifts. This helps validate the model's representations against clinical intuition and suggests that the adversarial training approach successfully creates meaningful counterfactual representations. Although the task of fully isolating causal effects in complex clinical text remains challenging, our findings highlight the potential of leveraging causal inference tools to guide more responsible and interpretable medical AI systems.

Ultimately, this work contributes to the growing interest in causality within natural language processing, encourages the application of causal reasoning to medical decision support, and lays the groundwork for future explorations of more nuanced interventions, broader datasets, and integration of additional medical knowledge sources.

References

Seppo Enarvi, Marilisa Amoia, Miguel Del-Agua Teba, Brian Delaney, Frank Diehl, Stefan Hahn, Kristina Harris, Liam McGrath, Yue Pan, Joel Pinto, Luca Rubini, Miguel Ruiz, Gagandeep Singh, Fabian Stemmer, Weiyi Sun, Paul Vozila, Thomas Lin, and Ranjani Ramamurthy. 2020. Generating medical reports from patient-doctor conversations using sequence-to-sequence models. In *Proceedings of the First Workshop on Natural Language Processing for Medical Conversations*, pages 22–30, Online. Association for Computational Linguistics.

Amir Feder, Nadav Oved, Uri Shalit, and Roi Reichart. 2021. CausaLM: Causal Model Explanation Through Counterfactual Language Models. *Computational Linguistics*, 47(2):333–386.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Preprint*, arXiv:1505.07818.

Seethalakshmi Gopalakrishnan, Luciana Garbayo, and Wlodek Zadrozny. 2024. Causality extraction from medical text using large language models (llms). *Preprint*, arXiv:2407.10020.

Yash Goyal, Uri Shalit, and Been Kim. 2019. Explaining classifiers with causal concept effect (cace). *CoRR*, abs/1907.07165.

Keith Harrigian, Ayah Zirikly, Brant Chee, Alya Ahmad, Anne Links, Somnath Saha, Mary Catherine Beach, and Mark Dredze. 2023. Characterization of stigmatizing language in medical records. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 312–329, Toronto, Canada. Association for Computational Linguistics.

André F. T. Martins and Ramón Fernandez Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. *CoRR*, abs/1602.02068.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alexander Wong. 2021. Umls-BERT: Clinical domain knowledge augmentation of contextual embeddings using the Unified Medical Language System Metathesaurus. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1744–1753, Online. Association for Computational Linguistics.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. 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 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

Jonathan G. Richens, Ciarán M. Lee, and Saurabh Johri. 2020. Improving the accuracy of medical diagnosis with causal machine learning. *Nature Communications*, 11(1):3923.

Ronald Xie, Steven Palayew, Augustin Toma, Gary Bader, and Bo Wang. 2024. WangLab at MEDIQA-M3G 2024: Multimodal medical answer generation using large language models. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, pages 624–634, Mexico City, Mexico. Association for Computational Linguistics.