Interpretability in NLP

A comparative analysis of two interpretability methods on medical text in light of the faithfulness metric

Bachelor Thesis

8th May, 2023

Raziye Sari

sari@cl.uni-heidelberg.de

Institut für Computerlinguistik

Ruprecht-Karls-Universität Heidelberg

Supervisor Prof. Dr. Anette Frank

Second assessor Prof. Dr. Katja Markert

Abstract

A central aim of the field of Explainable AI (XAI) is to provide *good* explanations for *black box* neural systems. One way this can be understood is in regards to *Faithfulness*: "How accurately does the explanation reflect the true reasoning process of the model? " With explanations depicting a model's inner-mechanisms through approximation, interpretations are at first potentially true and only with their evaluation gain validity. We probe this on a Sequence Classification task with two local post-hoc interpretability methods, namely Integrated Gradients (IG) and SHap Additive exPlanations (SHAP), that assign attribution scores to tokens of a given sequence. For this, we deploy the benchmarking tool *ferret*, that besides explanations from IG and SHAP, additionally offers their evaluation against *Faithfulness*. We find, that *ferret*'s IG constantly performs worse than SHAP, by assigning *unfaithful* attribution scores to tokens. Moreover, we observe IG assigning for specific types of tokens scores, that are totally contrasting those of SHAP and suspect the underlying reason for this in the "blind spot" problem of IG. Last but not least, we find that *ferret*'s *Faithfulness* metric exposes model (in-)security on label classes, that may be caused by imbalance in data.

Zusammenfassung

In dieser Arbeit werden zwei *post-hoc* Interpretationsmethoden aus dem Gebiet der Explainable AI (XAI), nämlich Integrated Gradients (IG) und SHapley Additive exPlanations (SHAP), unter Verwendung der *Faithfulness* Metrik gegenübergestellt. Beide Methoden liefern jeweils für Ausgaben (Output) eines Machine Learning (ML) Modells, Interpretationen auf Basis der repräsentativen Teile (Features) der Eingabe (Input). Die *Faithfulness* misst, wie gut diese Interpretationen die tatsächlichen Abläufe innerhalb eines ML Modells darstellen. Hierfür nutzen wir ein Tool namens *ferret*, das lokale Feature-Attributionsmethoden zum Interpretieren, sowie zum darauffolgenden Evaluieren bereitstellt. Unsere Objektiv wird in drei Schritten abgehandelt, wovon der erste die Bereitstellung von Ausgaben eines BERT Sprachmodells, genauer des *bert-base-german-cased*, angewandt auf der Sequenzklassifizierung ist. Wir führen eine multinomiale Klassifizierung von Sequenzen auf einem medizinischem Korpus (CARDIO:DE) mit elf Label-Klassen durch. Weiter werden diese Ausgaben von je beiden Methoden interpretiert, die den einzelnen Features des Inputs, den tokens, jeweils einen Wert auf einer Sklala von -1 bis 1 erteilen, repräsentativ für ihren jeweils positiven oder negativen Beitrag zu der richtigen Klassifizierung des Inupts. Je höher dieser Wert ist, desto höher wird der geleistete Beitrag für die Ausgabe der richtigen Klasse approximiert. Für den praktischen Vergleich der beiden Methoden, führen wir alz letzten Schritt die Evaluierung mit *ferret*'s *Faithfulness* auf Interpretationen von zehn zufällig gewählten Input-Output Instanzen durch. Ferret implementiert diese Metrik durch zwei "stress-test" Submetriken, die mittels bestimmter Modifikationenen des Inputs, die Stichhaltigkeit der durch die Methoden zugewiesenen Attributionen prüft. Diese sind *Comprehensiveness* und *Sufficiency*, wovon die erste, von der Methode als wichtig bewertete Features des Inputs löscht und zweitere, stattdessen nur ebenjene beibehählt. Es ist zu erwarten, dass das Aussortieren der positiv bewerteten Features (Comprehensiveness), sich negativ auf die richtige Klassifizierung auswirkt, während die auschließliche Inklusion dieser Features (Sufficiency), ausreichend für die Wahl der richtigen Klasse ist.

Nach Durchführung dieser Schritte gelangen wir zu folgenden Ergebnissen: Die Interpretationsmethode SHAP liefert in allen Label-Klassen wesentlich bessere Ergebnisse durch die *Faithfulness* Evaluierung, als IG (Tabellen 2, 3). Unsere Analyse zeigt, dass IG in 75% der Fälle von näher betrachteten Instanzen, bestimmten tokens nicht stichhaltige Attributionswerte zuschreibt (§6.2). Zu diesem Ergebnis gelangen wir, indem wir diese tokens individuell für die richtige

Klasse auswerten, sowie den Wert ihres Auftretens innerhalb ihres Kontexts durch Berufung auf SHAP's Interpretation aufzeichnen. Letztere Analyse zeigt ferner, dass IG diesen tokens einen Wert aus dem entgegengesetzten Bereich der Skala $[-1, 1]$ zu dem von SHAP gewählten Bereich zuweist, sodass ein negativer Attributionswert bei der anderen Methode ein positiver ist, und respektive in die andere Richtung (§5.1, §5.2). Diese Tendenz lässt IG sodann bemerkbar negativ herausstechen, weil SHAP, in einem Fall eines solchen tokens, diesen mit dem höchsten Wert innherhalb der Sequenz beurteilt (§5.2), in einem weiteren Fall eines tokens, dessen Aufnahme in die Sequenz (Sufficiency) den Klassifizierungswert wesentlich hinauftreibt (§5.2). Ferner stellen wir fest, dass IG in allen vier Fällen mehr tokens negative Werte zuschreibt als SHAP, und beweisen durch eine Zusatzstudie, dass die Inklusion dieser tokens in drei von vier Fällen, die Evaluierung der *Faithfulness* von IG verbessert (§6.2). Aufgrund der Wahl bestimmter Tokens durch IG verdächtigen wir, dass ein Problem mit der gewählten Baseline für diese Methode vorliegt (6.2), betonen aber auch, dass unser Erklärungsversuch $2/3$ der Fälle abdeckt und eines außen vor lässt.

Erwähnenswert ist auch, dass die *Faithfulness* Evaluierung, wie sie bei *ferret* durch *Comprehensiveness* und *Sufficiency* vorzufinden ist, Einsicht verleiht in die (Un-)Sicherheit des Modells auf den jeweiligen Label-Klassen, welche die unterschiedliche Datenverteilung zur Ursache hat. Dies schließen wir aus der kontrastierenden Performanz einer Klasse (§ 6) auf *Comprehensiveness* und *Sufficiency*.

Wir bemerken zuletzt, dass wir unsere Implementierungen öffentlich zugänglich machen[1].

1 `https://github.com/dieterich-lab/xai-in-nlp`

Contents

List of Figures

List of Tables

1 Introduction

The field of Explainable AI (XAI) aims to understand why an Artificial Intelligence (AI) system produces a specific output [Danilevsky et al., 2020]. A simple example from NLP in a classification task would be: Given the input sequence "The food here tastes delicious" and a model classifying it rightly with the sentiment label "Positive", methods used in XAI unravel the reasoning process of the model which made it arrive at its choice of output [Attanasio et al., 2023]. For this, a plethora of diverse interpretability methods yielding different types of explanations have been set forth [Danilevsky et al., 2020]. An interpretability method for the prior example for instance, can produce an explanation, disclosing how much each word in the input sequence contributed to the prediction of "Positive". Given a well performing model, the word "good" should be highlighted as contributing majorly [Attanasio et al., 2023]. Through these efforts, the system as well as the data it has seen during development can be better understood and improved [Sundararajan et al., 2017].

Furthermore, with interpretability methods providing such and other forms of insight into the reasoning process of an AI system and thus enabling a deeper assessment of its sanity, a potential use-case in public institutions can be thought about. However, especially in sensitive fields like those of health and law, where decisions carry a lot of weight, the deployed algorithms aiding professionals in their work are required to be as trustworthy as possible. One such example of a proposed application is for *Diabetic Retinopathy Prediction*, for which the system Automated Retinal Disease Assessment (ARDA) has been developed to aid authorities of health [Bora et al., 2021]. While concerns about the use of Machine Learning (ML) in such settings regarding ethics have been vocalized [Char DS, 2018], other more technical issues relating to the validity of the model should and are ideally resolved beforehand. Bias in data constitutes a reasonable problem in environments like those, to name one example [Char DS, 2018]. If the data with which the model was fed is discriminating against a group of people based on their heritage, the model will reflect that in its decisions [Novak et al., 2023]. A good interpretability method should yield explanations for the output revealing such critical areas, at best to the developer.

Following the best case scenario, that an explainer provides explanations through which it can be agreed upon, that the model predicts in a sane way, a further pressing question is: "How accurately does the explanation *actually* reflect the true reasoning process of the model? " [Jacovi and Goldberg, 2020]. In the according literature this is referred to as *Faithfulness* and plays an

essential part in defining what a good interpretability method should fulfil [Harrington et al., 1985]. Following the above example of *Diabetic Retinopathy Prediction*, an unwanted scenario would be that a doctor is faced with an unfaithful explanation of a prediction, such that the model based its decision on other factors of the input, deviant from those highlighted by the explainer. What could further worsen such a case is if the explanation appears convincing to the doctor, making it less likely that they will question the model's accuracy. Hence, *Plausibility* [Wiegreffe and Pinter, 2019] describing how convincing the explanation is to humans constitutes besides *Faithfulness* another metric for the evaluation of explainers.

With an overview given on the tasks and challenges of interpretability methods of XAI, the following work concentrates on interpretability in the NLP domain, concretely on the task of multi-labelled sequence classification. In this study, we analyse and compare two interpretability methods, namely Integrated Gradients (IG) [Sundararajan et al., 2017] and SHapley Additive exPlanations (SHAP) [Lundberg and Lee, 2017] in light of the *Faithfulness* metric from *ferret* [Attanasio et al., 2023]. Through this, we aim to give insight into possible reasons one might perform better than the other. In efforts to this, the following chapter, Related Work, provides an overview of the current research on XAI in the NLP domain. The chapter Methods after that includes a comprehensive introduction into both mentioned post-hoc interpretability methods deployed in the remainder of this work. In Experiments, the procedural steps taken to training a language model as well as integrating *ferret*, a benchmarking tool for generating explanations and evaluations are shared. We make our code publicly available[1]. In the Results chapter, the outcomes of the previous experiments are disclosed as stand-alone model performance on label classes and in more focus, the evaluation scores for the explanations. Following that, the chapter Discussion analyses the results from before based on differences between both interpretability methods and label-specific performances, whilst also pointing out more salient observations. Finally, the findings are concisely reiterated in the Conclusion.

1 https://github.com/dieterich-lab/xai-in-nlp

2 Related Work

In the field of Natural Language Processing (NLP), Danilevsky et al. (2020) present a survey summarizing and organizing recent works on Explainable AI for NLP. In it, they propose a categorization of methods for interpreting NLP models, such that methods can either be *local* or *global* explainers and further, be of *self-explaining* or *post-hoc* type. The former differentiation regards the scope of the explanation, by which *global* ones explain the whole model as a predictor, whereas *local* approaches concentrate on explaining single input-output pairs. Their survey shows that the majority number of works make use of the local approach. This might seem surprising, since white-box-systems like ones based on rules, decision trees, hidden Markov models and logistic regressions have a solid foundation in the field of NLP and are globally explainable [Danilevsky et al., 2020]. The authors however, posit that opaque models are harder to untangle and because of this, stay in the foreground of research. The latter distinction of *self-explaining* versus *post-hoc* focuses on whether the model is used as an explainer for itself or additional methods outside the model are utilized, respectively. For the first category of self-explanation, the attention mechanism of Deep Neural Networks (DNN's) has seen wide use in NLP tasks over time [Danilevsky et al., 2020]. This is because the attention weights of features represent the amount of "focus" the model puts on them during the prediction [Bahdanau et al., 2016] and are easily obtained by backpropagation [Vaswani et al., 2017]. Each input feature is assigned an importance score, which can be visualized on a graded scale in so called heat-maps. Besides attention, another widely used operation enabling *explainability* is first derivative saliency, as proposed by Simonyan et al. [2014] by which the partial derivative of the output is computed with respects to the input [Samek et al., 2017]. This approach constitutes to Gradients [Simonyan et al., 2014], which is outrun by its successor Integrated Gradients [Sundararajan et al., 2017], currently among the state-of-the-art post-hoc explainers [Attanasio et al., 2023]. Danilevsky et al. [2020] find that more than half the regarded works on XAI make use of feature importance based explainability, with both mentioned operations on the forefront.

Besides the mere explanations, Jacovi and Goldberg [2020] also point to the issue of evaluating those explanations and stress that many works provide only informal evaluation strategies, which is a consequence of missing standardized metrics. The authors contribute to solving this by organizing the present evaluation strategies and pointing out that many of research give evaluations based on human intuition. However, other operation-specific techniques, in this case regarding feature

importance explainability, have also been proposed in recent years. Serrano and Smith [2019] for example set the attention weight of the highest attributed feature to zero and watch for changes in the output. While this perturbation effects the inner-architecture of the model, similar "stress-tests" are also probed on the input, by erasing those features, that according to post-hoc methods are most/least important to the output [DeYoung et al., 2020].

Apart from *how* the explanation is carried out, the concrete definition of *what* is being explained is also discussed in the literature. In agreement with Danilevsky et al. [2020], Jacovi and Goldberg [2020] also underline that many works do not draw distinct lines around the specification of what they are evaluating given the explanation. They also note, that works conflate the two terms *Faithfulness* and *Plausibility* [Harrington et al., 1985], [Wiegreffe and Pinter, 2019], [Jacovi and Goldberg, 2020], which Danilevsky et al. [2020] define as *fidelity* and *comprehensibility*, respectively. While the former is concerned with how accurate the explanation reflects the true reasoning of the model, the second one regards whether the explanation is sensible to a human. This distinction is important since cases where one but not the other is fulfilled can appear, as described on the context of *Diabetic Retinopathy Prediction* (Section 1). To align the literature on *Faithfulness* along standardized axes, Jacovi and Goldberg [2020] propose a set of three assumptions. Besides the *Model Assumption* and *Prediction Assumption*, the third, *Linearity Assumption*, states that heat map interpretations are faithful under certain circumstances, given that contributions of the parts of the input are independent from each other.

Discussions are also held on the concept of *Faithfulness*. It is argued that methods being a mere approximation of the actual inner-workings of a system, are as a result not in the state to provide an exact portrayal of the explained model [Jacovi and Goldberg, 2020]. For the second, Jacovi and Goldberg [2020] highlight that many works approach *Faithfulness* evaluation via counter-example, by proving that a method is *not faithful*. The authors consider this unproductive, rather calling for a graded assessment on the *Faithfulness* of a method by proposing the term "sufficiently" *faithful*. This terminology in practice can be understood twofold, where, in the first case, the degree of *Faithfulness* of a method is graded across models and tasks whereas for the second, it is judged across the input space, such that for specific parts of the input, the same method might be considered more *faithful* than for other parts [Jacovi and Goldberg, 2020].

Regarding the above argument about the gap between the actual mechanisms inside a system and that which a method is able to grasp from them, Sundararajan and Taly [2018] state, that

those features of the input that share the value of the baseline are expected to be assigned zero attribution by IG. Sturmfels et al. [2020] take this statement further by visualizing that the choice of baseline is decisive in the quality of the generated explanation.

With some current and central discussions regarding efforts in XAI, and more specifically for the field of NLP shared, and IG [Sundararajan et al., 2017] mentioned, the following chapter provides a more in-depth description for the former and SHAP [Lundberg and Lee, 2017] as two state-of-the-art post-hoc interpretability methods.

3 Methods

In the following, IG [Sundararajan et al., 2017] and SHAP [Lundberg and Lee, 2017] are introduced as two post-hoc interpretability methods, yielding local explanations.

3.1 Integrated Gradients

[1] Sundararajan et al. [2017] introduce IG as a follow-up method to its predecessor Gradients [Simonyan et al., 2014].

Gradients of the prediction score of a model with respects to the input approximates for deep non-linear models like DNN's the feature importances. It is for a model $M(x)$ defined as: Given a baseline and an input feature of the model, calculate the gradient of the feature vector at the given input. This yields an importance score, representing the importance of that feature value to the model. Respective of this, the gradient of the baseline needs to be zero, hence representing the null output of the model. However, this approach of calculating gradients reaches its limits once confronted with edge cases: Given a model as

$$M(x) : 1 - max(0, 1 - x)$$

which processes the input feature of value x with a ReLu activation function, the gradient function then takes the form:

$$G(M, x) : max(0, sign(1 - x) \cdot x).$$

Calculating the baseline input value of 0 with the gradient function yields: $G(M, 0) : 1 \cdot 0 = 0$, hence an importance score of 0. But with an input feature value of 2, the gradient function again scores: $G(M, 2) : 0 \cdot 2 = 0$, the same score as the one for the baseline. In fact, every other value above 2 will also have a score of 0, saturating over the function. Hence, according to the gradient method, these input feature values are *not important*, since they yield the same value as the baseline of the model. This counters the intended expression of a feature attribution method,

1 As obtained through "Feature Attribution | Stanford CS224U Natural Language Understanding | Spring 2021." Youtube, Professor Christopher Potts, 07.01.22, `https://www.youtube.com/watch?v=RFE6xdfJvag&t=221s`

which is to reflect the change in model output due to a modification to the feature value. The authors Sundararajan et al. [2017] set this characteristic as a required property of every feature attribution method and show that IG fulfils this.

IG likewise takes a baseline and an input feature for a model $M(x)$, but unlike its predecessor, accumulates the gradients at steps along the straight-line path. This is formulated as:

$$IG(M, x, x') = (x_i - x_i') \cdot \sum_{k=1}^{m} \frac{\partial M(x' + \frac{k}{m} \cdot (x - x'))}{\partial x_i} \cdot \frac{1}{m},$$

where x' denotes the baseline and m the maximum amount of steps. With this approach, for the same input feature value $x = 2$ and the given baseline value $x' = 0$, the explanation function for the above model results in:

$$IG(M, 2, 0) = (2 - 0) \cdot \sum \begin{pmatrix} max(0, sign(1 - 0.0) \\ max(0, sign(1 - 0.2) \\ max(0, sign(1 - 0.4) \\ \vdots \\ max(0, sign(1 - 2.0) \end{pmatrix} \cdot \frac{1}{m} \approx 1. \tag{1}$$

Through this approach, an importance score that is different from that of the baseline is arrived at. As noted for Gradients before, IG fulfils an essential requirement for attribution methods.

The authors introduce three axioms, that attribution methods should fulfil, of which first, *Sensitivity* is defined as: "If for every input and baseline that differ in one feature but have different predictions, then the differing feature should be given a non-zero attribution." [Sundararajan et al., 2017]. As illustrated above (Eq. 1), IG satisfies this proposition. The second desirable characteristic *Implementation Invariance*, states that the attributions of any attribution method should always be identical for two functionally equivalent networks. Such two networks are functionally equivalent, if for all inputs the outputs of both are identical. Because gradients are invariant to implementation due their application of the implementation invariant chain-rule, this property is also fulfilled for IG [Sundararajan et al., 2017]. Lastly, the authors define *Completeness*, which proposes that attributions should add up to the difference between the output of M at the input x and the baseline x'. Calculated with the above model as: $M(x = 2) = 1$ and $M(x' = 0) = 0$, the importance score of the feature value (Eq. 1) equals the difference in the outputs, thus making IG *complete*.

3.2 SHapley Additive exPlanations

[2] SHAP is an additive attribution method as introduced by Lundberg and Lee [2017] and is based on Shapley values from cooperative game theory [Lipovetsky and Conklin, 2001].

Shapley Values is a means of determining the share of a value among parties that contributed to that value. Formally, given a coalition c with members $m \in c$ inside this coalition that produce a final value v, Shapley Values calculates how much each member $m \in c$ contributed to that value v. The steps of the determination process are as follows:

1. Sample all possible coalition pairs of type $(x_m, y_{\setminus m}) \in C$, where x, y denote two coalitions with the *only* difference, that the member of interest m is included in one and missing in the other.

2. Calculate for each possible pair the difference in produced value $v_{diff} = v_x - v_y \ \forall (x_m, y_{\setminus m}) \in C$. This is referred to as the marginal contribution of m.

3. Finally, taking the mean over all these marginal contributions, yields the Shapley Value for member m.

SHAP reformulates the above determination for share of value as a linear regression problem, through which it becomes applicable to ML models:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j \tag{2}$$

where g denotes the explanation model (based on the original model f) and z' a simplified version of the input x. ϕ_0 is the average output of f and ϕ_j represents the feature importance for feature j. M is the maximum coalition size. The simplified version of the input x is at basis a coalition vector z' of 0's and 1's, where 0 represents the exclusion of a feature and 1 the inclusion respectively. Where with four features, the number of all possible coalitions is kept in a feasible range, this surges dramatically when stepped up further, e.g. more than 17 Billion possible coalitions with 32 features. In order to bypass this problem Lundberg and Lee [2017] propose KernelSHAP, a means of approximating Shapley Values through permutation samples and weighted linear regression.

---

2 As obtained through "Shapley Additive Explanations (SHAP)." Youtube, Rob Geada, 08.06.21, `https://www.youtube.com/watch?v=VB9uV-x0gtg&t=415s`

It is not rare, that most DNN's work with more than 32 features, for which interpreting their outputs through SHAP only becomes feasible through approximation, in this case described for KernelSHAP. It is formulated as: For a given input z, sample coalitions $z' \in \{0,1\}^M$ by replacing the features either with 1's or in case of missing features (0's), permuting the feature value from background data B. Now similar to the consideration of differences in value (item 2), the model output for a coalition is needed. In order to transform the altered coalition into a valid input, it is run through a mapping function in the form $h_x(z') = z$, which turns 1's back into original feature values and 0's are translated into permutation values. These permuted samples are then included as samples for the weighted linear regression model.

$$L(f, g, \pi_{x'}) = \sum_{z' \in Z} [f(h_x(z')) - g(z')]^2 \ \pi_{x'}(z')].$$

Through which, squared error loss is computed using the normal output of the original model and deducing from it the explanation model output of the simplified coalition vector. This term is then weighted for each coalition $z' \in Z$ with the following formula:

$$\pi_{x'}(z') = \frac{(M-1)}{(M \ choose \ |z'|) \ |z'| \ (M - |z'|)}$$

where the denominator from left to right considers: the number of coalitions the same size as $|z'|$, the number of non-zero, as well as the number of zero elements in z'. This is to weight those coalitions stronger, that comprise of either few 1's (many zeroes) or many 1's (few zeroes). This is because they provide insight into the isolated effects of features, which naturally is of high interest for a method, that ascribes importance scores to features individually. Through this approach of sampling coalitions, permuting them accordingly and weighting each with respects to their amount of features contributing to the output, a close approximation of feature importance is achieved [Lundberg and Lee, 2017].

With insight given on most central aspects of both post-hoc interpretability methods, IG and SHAP, the following chapter gives details about the conducted experiments.

4 Experiments

In the previous chapter, IG and SHAP are introduced as two post-hoc interpretability methods. Now, the experiments are disclosed in the following manner: First, we deploy a model on a multinomial Sequence Classification task, yielding input-output pairs. Those pairs are then explained with *ferret*'s IG and SHAP explainers. The resulting explanations are finally evaluated on the basis of *ferret*'s implementation of the *Faithfulness* metric.

4.1 Sequence Classification Task

The Sequence Classification task for NLP takes as input to a model a sequence of text and predicts for it a label class out of two or more total possible classes, with the former describing binary and latter a multinomial setting respectively. We use the pre-trained BertForSequenceClassification, specifically *bert-base-german-cased* from the huggingface panel as our BERT model [Devlin et al., 2019] with its corresponding tokenizer.

BERT model. BERT is a large pre-trained language model [Devlin et al., 2019]. Its name stands for "Bidirectional Encoder Representations from Transformers". BERT is pre-trained to understand language, moreover the context in that language through two objectives, namely Masked Language Modelling (MLM) and Next Sentence Prediction (NSP) using transformer encoders [Vaswani et al., 2017]. While for the MLM objective, BERT aims to correctly identify masked out parts of the input similar to "fill in the blanks", the task of NSP lies in deciding whether the next sentence of the input contextually follows from the first or not. During pre-training of BERT, it is fed with text sequences, which are preprocessed to be in the form of input embeddings. Each token in the sequence is processed threefold. For one, TokenEmbeddings are generated by splitting up the text into smaller parts, called tokens, using WordPiece embeddings [Wu et al., 2016] from a 30,000 token vocabulary. In addition to that, each token receives an identifier, representing which sentence it belongs to in the sequence, namely SegmentEmbeddings. Through this, the first sentence is distinguished from the second, important for the NSP objective. Finally, with PositionEmbeddings, each token is assigned a number locating its position in the input. The processed text is then passed through BERT's stack of transformer encoders, comprising

Label	Precision	Recall	F1-Measure	Support
Anrede	**1**	**1**	**1**	99
Medikation	0.99	0.98	0.98	1627
KUBefunde	0.99	0.97	0.98	1105
Diagnosen	0.96	0.97	0.96	1738
AllergienUnverträglichkeitenRisiken	0.97	0.94	0.96	236
Abschluss	0.94	0.99	0.96	2472
Befunde	0.93	0.86	0.9	2519
Zusammenfassung	0.9	0.9	0.9	2138
Anamnese	0.9	0.81	0.85	1097
Mix	0.76	0.83	0.79	242
EchoBefunde	0.6	0.94	0.73	290

Table 1: Test Results - sorted in descending order of f1-measure - when tested on 100 doctoral letters. Bold markings show best and orange markings show worst results.

of 12 layers for BERT base. The input is processed simultaneously, for which BERT generates two outputs, one for each trained objective: A binary classification whether the second sentence follows from the previous (NSP), as well as the original input with predicted words for those that were masked out (MLM).

CARDIO:DE corpus. The data used for the multinomial Sequence Classification task is the CARDIO:DE corpus from the Dieterich Lab in Heidelberg [Richter-Pechanski et al., 2023]. It consists of 500 manually annotated doctoral letters from the cardiology department of Heidelberg University Hospital from the time of 2020-2021. In each letter, any contained personal information regarding patients, professionals etc., that correspond to Protected Health Information (PHI)[1] identifiers are de-identified through replacement with semantic placeholders. However, week days, patient age, company names and names of products are kept. Each section of the discharge letter is annotated to belong to one of 11 label classes: *Anrede, Diagnosen, AllergienUnverträglichkeitenRisiken, Anamnese, Medikation, KUBefunde, Befunde, EchoBefunde, Zusammenfassung, Mix* or *Abschluss*.

We fine-tune the pre-trained BertForSequenceClassification on sequences derived from 400 such doctoral letters, namely CARDIO:DE400. The training data is split into 90 % train and 10 % development sets. We set the tokenizer to have a maximum sequence length of 128 in order to train on two GPU's for 2 epochs, with the batch size set to 5.

1 `https://www.hipaajournal.com/considered-phi-hipaa/`, last visited: May 5th, 2023

The final model has an accuracy score of 93 % when tested on the remaining sequences from the 100 held-out letters, CARDIODE:100. Table 1 showcases the results per label with precision, recall and f1-measure scores and the amount of classified sequences (support). *Anrede* is the only label with perfect scores for all three metrics, whereas *EchoBefunde* scores worst for precision and F1-measure, with *Anamese* performing worst for recall.

We then randomly choose 10 input sequences per label that are classified correctly by our model to explain the predictions using *ferret*.

4.2 Ferret

Attanasio et al. [2023] introduce *ferret* as a tool for benchmarking diverse interpretability methods on transformer models, by offering two main functionalities. For one, it generates *explanations* with one of six interpretability methods, among them SHAP and IG. These explanations contain for each token in the input the attribution score calculated with the method. The higher the score, the more important it is for the model to predict the given label. Secondly, it evaluates the produced explanation against the *Faithfulness* metric, hence "how accurately it reflects the true reasoning process of the model" Jacovi and Goldberg [2020]. For this purpose, the authors mention two sub-metrics, namely *Comprehensiveness* and *Sufficiency* [DeYoung et al., 2020] as ways of conducting "stress-tests". Through them, the input is modified to exclude (*Comprehensiveness*) or only include (*Sufficiency*) important tokens of the sequence [DeYoung et al., 2020]. Since as expected, most important tokens of the input - as determined by the interpretability method - drive the prediction more than those that are least important, both sub-metrics test whether the a method's *claims* about the tokens are upheld.

Explaining with ferret. Given 10 pairs of input sequence and correct prediction of our model, the explainer, one of IG or SHAP, takes in a pair of input sequence and prediction - here corresponding to the gold label - and outputs a list of the included tokens, where each one is assigned an importance score, representative of its contribution to the correct prediction. While 1 denotes *importance*, assigned to input features that are judged to contribute positively to the right prediction, -1 represents *unimportance*, thus assigned to negatively contributing tokens. A sample explanation from *ferret* is illustrated in the Appendix (Table 4).

Evaluating with ferret. With ten explanations generated by IG and SHAP, the next step includes evaluating these with *ferret*'s *Faithfulness* metric. *Faithfulness* for an interpretability method is defined as: "How accurately does the explanation actually reflect the true reasoning process of the model? " [Jacovi and Goldberg, 2020] and was prior mentioned in the field of mathematics [Harrington et al., 1985]. Jacovi and Goldberg [2020] mention *erasure*-based tests, through which a method is deemed *faithful*. *Ferret* implements both *Comprehensiveness* and *Sufficiency* [DeYoung et al., 2020] as two sub-metrics for that purpose. Since they consider parts of the text that are then either truncated or solely kept, some important specifics about their deployment in *ferret* are disclosed in the following.

To evaluate the explanation at hand with either *Comprehensiveness* or *Sufficiency*, *ferret* first filters out tokens that have a negative attribution score. Then, with steps of ten in the range $[10, 100]$, the $k\%$ most important tokens are selected from the input text, resulting in a set r. Consequently, the last step of $k = 100\%$, contains all tokens with a positive score. For each step, the *Comprehensiveness* score for the modified sequence is calculated as:

$$Compr = f(x)_j - f(x \setminus r_j)_j$$

where $f(x)_j$ denotes the prediction score for the sequence x for label j, while $x \setminus r_j$ represents the sequence x modified to exclude those tokens, that are in r. Once the individual scores for the steps are calculated, the mean over them is the final *Comprehensiveness* score for that explanation. *Sufficiency*, at basis works in the same way with the only difference of solely keeping, rather than removing the selected tokens $\in r$:

$$Suff = f(x)_j - f(r_j)_j.$$

Scores are judged to be better if higher for *Comprehensiveness* and lower for *Sufficiency*.

We note here, that *ferret's* modification to the input text is practised through deletion of the respective tokens, rather than a replacement with a placeholder, e.g. a special token of the tokenizer (for example `"[MASK]"`). In the NLP setting, this leads to malformed text, which the tokenizer nevertheless still splits into the same tokens.

With ten explanations for each label evaluated on *Comprehensiveness* and *Sufficiency*, the mean of the ten scores acts as a representative per label in the following chapter sharing the results.

5 Results

In the following, the *Comprehensiveness* and *Sufficiency* evaluation scores for the explanations are shared. This is done by regarding the best and worst performing labels for SHAP per sub-metric, then comparing to it IG' performance.

5.1 Comprehensiveness

Table 2 shows the mean Comprehensiveness scores over ten explanations per label, evaluated for IG and SHAP. The label *Anrede* scores perfect for *Comprehensiveness* with SHAP's provided explanations, whereas *Abschluss* is the worst-scorer respectively. IG's best scoring label is *AllergienUnverträglichkeitenRisiken*, while *Anamnese* takes the last place. This raises the question, why some labels perform better than others. Further, whether there's a correlation to the scores of the labels on the given task, though no immediate correlation to the f1 score is present, since Comprehensiveness' second best-scorer *Mix* for SHAP (IG' best-scorer) only takes the penultimate position for f1-measure (table 1).

Further observation shows that IG and SHAP are assigned scores in separate intervals, such that the former method's record of best scored label is approximately in line with SHAP's worst scored label.

***Anrede* scores best with SHAP.** *Anrede* receives a perfect mean score ($Compr = 1.0$) with the explanations provided by SHAP, meaning all ten of the explained samples were evaluated to being $Compr \geq 0.99$. It is also the best-scoring label for the Sequence Classification task with regards to all three metrics (table 1). To observe the evaluation closer, the following sample is given.

$f(x)_j = 1.0$, $j =$ *Anrede* and $x =$ "über Ihren Patienten B-SALUTE B-PER I-PER geboren am ⟨[Pseudo] 24/06/1977⟩ wohnhaft in B-PLZ B-LOC I-ADDR I-ADDR der sich vom bis in unserer stationären Behandlung befand.".

Label	Mean Scores SHAP	IG	F1-Score
Anrede	**1.0**	0.4	**1.0**
Mix	0.86	**0.64**	0.79
AllergienUnverträglichkeitenRisiken	0.85	0.31	0.96
KUBefunde	0.77	0.3	0.98
Diagnosen	0.75	0.31	0.96
Zusammenfassung	0.75	0.14	0.9
Befunde	0.67	0.2	0.9
EchoBefunde	0.66	0.25	0.73
Anamnese	0.64	0.02	0.85
Medikation	0.64	0.13	0.98
Abschluss	0.61	0.3	0.96

Table 2: Comprehensiveness mean scores for SHAP & IG with F1-scores per label

SHAP's explanation for this sample is evaluated with a mean *Comprehensiveness* score of:

$$Compr^{\text{SHAP}}_{\textit{Anrede}} = f(x)_j - f(x \setminus r_j) = 1.0$$

where $f(x)_j$ denotes the prediction probability for the given label j, hence a 100% probability score for *Anrede*. The 10% tokens chosen in the first step, that contribute most to the prediction according to SHAP are: $r_{10} =$ [über$_{0.27}$, Ihren$_{0.16}$, Patienten$_{0.09}$, befand$_{0.04}$], with each token assigned its attribution score in subscript. Through removal of these tokens, the sample transforms to:

$$f(x \setminus r_{10})_j = 0.0,\ x = \texttt{"B-SALUTE B-PER I-PER geboren am <[Pseudo] 24/06/1977> wohnhaft in B-PLZ B-LOC I-ADDR I-ADDR der sich vom bis in unserer stationären Behandlung."}$$

where the 100% drop in prediction probability for *Anrede* shows that these tokens, when included did contribute to the prediction of the right label, and the lack thereof constitutes to the prediction of a false label, namely *Abschluss* with a probability of $f(x \setminus r_{10})_{Abschluss} = 98\%$. The remaining nine steps stay in this scoring range $Compr_k >= 0.99, k \in [20, 100]$, meaning the gradual increase of removed tokens r does not influence the probability for the gold label any further. Hence, the removal of four tokens in the first step has the single biggest effect. While this is true for the label class *Anrede*, the prediction of a false class in each step is either *Abschluss*, *Zusammenfassung* or *Anamnese* with varying scores in the range $[0.32, 0.98]$.

***Anrede* with Integrated Gradients.** The mean Comprehensiveness score for the above sample with the explanation provided by IG is:

$$Compr^{\text{IG}}_{\textit{Anrede}} = f(x)_j - f(x \setminus r_j) = 0.51$$

which is around half the score that SHAP's explanation receives. With IG's explanation, *ferret* chooses only in the second step the same amount of tokens as is did with SHAP's explanations in the first step. The reason behind this is in IG assigning more negative attributions to tokens, that are then filtered out. Hence for $k = 20\%$ the most important tokens are $r_{20} =$ [über$_{0.37}$, Ihren$_{0.04}$, L$_{0.03}$, B$_{0.01}$] resulting in the sample:

$f(x \setminus r_{20})_j = 0.37$, $x =$ "Patienten B - SAUTE B - PER I - PER geboren am < [Pseudo] 24 / 06 / 1977 > wohnhaft in - PLZ B - LOC I - ADDR I - ADDR der sich vom bis in unserer stationären Behandlung befand.".

Through a different choice of tokens omitted from the text, the probability for *Anrede* does not drop as significantly as it did through SHAP's explanation, such that a decrease of 63% is recorded in comparison to a prior loss of 100%. The lesser effect on the probability for the gold label, for IG leads to a majority number of modified sequences throughout the steps to being classified correctly as *Anrede*, reaching a pinnacle in the sixth step: $f(x \setminus r_{60})_{Anrede} = 0.7$.

Interim Conclusion. Detailed insight into the evaluation of both IG's and SHAP's explanations for *Anrede* shows that the choice of tokens from IG's explanation results in a smaller change of the prediction probability for the correct label class. SHAP on the other hand, selects tokens that once omitted, undermine the score more.

***Abschluss* scores worst with SHAP.** The label *Abschluss* is scored worst with SHAP's explanations ($Compr = 0.61$). On the Sequence Classification task, its recall score comes second with its precision as well as f1 score located in the top 15% for all labels (Table 1). With a sample evaluation given as:

$f(x)_j = 1.0$, $j =$ *Abchluss* and $x =$ "I-PER/I.",

the prediction probability is like the sample of *Anrede*, 100% for the gold label. Because of the shortness of the sequence text, the procedure of selecting top $k\% \in [10, 100]$ of the tokens is limited to a number of steps that take in at least one new token into the set of tokens r. This is because SHAP's explanation here includes less than 10 positively attributed tokens. *Ferret* does not count in steps of those r_k in the mean score that do not grow with regard to the predecessor r_{k-1}, thus duplicate steps with same r_k are left out from the final score. Consequently, the selection of r for this sample starts at $k = 20$:

$$f(x \setminus r_{20})_j = 0.7,\ x = \texttt{"I-ER/I."},$$

with $r_{20} = [\texttt{P}_{0.44}]$. Prior to this on the account of *Anrede*, the sample loses 98% of its probability for the right label, once four tokens are removed from the sample. Here, the exclusion of one token has a relatively similar effect on the gold label probability, such that the token's abandonment leads to a reduction of 30%, which reaches zero probability in the immediate next step of $k = 30\%$, staying constant until the last step of $k = 100$:

$$\begin{gathered} f(x \setminus r_{30})_j = 0,\ x = \texttt{"-ER/I."} \\ \vdots \\ f(x \setminus r_{100})_j = 0,\ x = \texttt{"ER"}, \end{gathered}$$

with $r_{30} = [\texttt{I}_{0.26}, \texttt{P}_{0.44}]$ and $r_{100} = [\texttt{I}_{0.26}, \texttt{-}_{0.03}, \texttt{P}_{0.44}, \texttt{I}_{0.07}, \texttt{.}_{0.04}]$. The remaining sequence text with positive attribution tokens truncated in the final step, exposes those tokens, that got filtered out for being negatively attributed. For this instance `"ER"` is one such token. This left out token is important because of its contrary role in IG's selection of r.

***Abschluss* with Integrated Gradients.** For the same sample, IG chooses `"ER"` to be the most important token. The sample therefore in the second step is:

$$f(x \setminus r_{20})_j = 0.99, x = \texttt{"I-P/I."}$$

with $r_{20} = [\texttt{ER}_{0.52}]$. As apparent from the resulting score of $Compr_k = 0.01, k = 20$, this exclusion does not effect the probability in a similar way to the token choice through SHAP before. Quite contrarily, the exclusion of `"ER"` up until the last step of:

$$f(x \setminus r_{100})_j = 0.88, x = \texttt{"I-/."}$$

with $r_{100} = [\texttt{P}_{0.04}\texttt{ , ER}_{0.52}\texttt{ , I}_{0.09}]$ results in a sequence, that is by 18 score points classified more confidently as *Abschluss* by the model than the minimally altered sequence through SHAP's explanation in the first step.

Concluding Comprehensiveness. Regarding *Comprehensiveness* closely, gives the insight that SHAP provides more *faithful* explanations according to *ferret's* computation. Moreover, IG assigns more negative values to tokens and may assign the highest importance to a token, which receives negative attribution with SHAP. It is left to explore these findings in the following section for *Sufficiency*.

5.2 Sufficiency

Table 3 shows the mean *Sufficiency* scores per label, evaluated for both methods. In alignment with *Comprehensiveness*, IG for *Sufficiency* also performs worse than SHAP in all labels, with its best scorer *EchoBefunde* sharing the score of SHAP's worst scorer *Mix* ($Suff = 0.37$). *Mix* is also in last place through IG's explanations ($Suff = 0.8$). This agreement of both methods prompts the analysis - against previous statement about the absence of an immediate correlation (Section 5.1) - whether and in the case of, to what degree the model performance may be reflected in these metrics. Equal to *Anrede*'s score for *Comprehensiveness*, *Zusammenfassung* scores perfect for *Sufficiency* with SHAP's explanations ($Suff = 0.0$).

***Zusammenfassung* scores best with SHAP.** *Zusammenfassung*, like *Anrede* for *Comprehensiveness*, scores perfect with SHAP's explanations for *Sufficiency*. Given the mean *Sufficiency* score and the sample:

$$Suff = f(x)_j - f(r_j)_j = 0.0$$

$f(x)_{Zsf} = 1.0$ and $x =$ `"Röntgenologisch wurde der V.a. eine Stauungspneumonie gestellt."`,

the first step of the calculation results in:

$$f(r_{10})_{Zsf} = 0.99,\ x = \texttt{"wurde"}.$$

Label	Mean Scores SHAP	IG	F1-Score
Zusammenfassung	**0.0**	0.44	0.9
Befunde	0.02	0.41	0.9
Anamnese	0.03	0.46	0.85
EchoBefunde	0.04	**0.37**	0.73
AllergienUnverträglichkeitenRisiken	0.06	0.6	0.96
Medikation	0.07	0.44	0.98
Anrede	0.1	0.71	**1.0**
Abschluss	0.15	0.45	0.96
Diagnosen	0.19	0.66	0.96
KUBefunde	0.25	0.77	0.98
Mix	0.37	0.8	0.79

Table 3: Sufficiency Mean scores for SHAP & IG with overall F1-scores for each label

With the sole inclusion of one token from the sequence, the classification score of the original sample is almost mirrored ($f(x)_{Zsf} = 1.0$), reaching the same score along the steps:

$$f(r_{100})_{Zsf} = 1.0,\ x = \texttt{"ologisch wurde der einepneonie gestellt."},$$

The issue about unintelligibility in text as a consequence of truncating tokens is observable here, where through concatenating tokens to each other, words are created that don't exist. Nevertheless, because the sequence is tokenized, the model only sees a different combination and ordering of tokens, that is new. For this altered version, the model puts all probability on the gold label.

***Zusammenfassung* with Integrated Gradients.** For the same sample, IG's explanation initially creates the sample:

$$f(r_{10})_{Zsf} = 0.0,\ x = \texttt{"Röntgen"}$$

where the inclusion of a single different token, has the total opposite effect on the prediction probability: *Zusammenfassung* loses all probability and instead *Befunde* is predicted with an approximately equal probability to $f(r_{10})^{SHAP}_{Zsf} = 0.99$, that is $f(r_{10})_{Bfd} = 0.96$. This remains constant up until the last step $k \in [20, 100]$, such that the sequences are always classified as *Befunde* ($f(r_k)_{Bfd} >= 0.96$). The token `"wurde"` gets negative attribution from IG, thus is filtered out from

the beginning, which is contrary to SHAP's decision to mark it as highest contributing. This decision has a leading impact throughout the steps, where in the last of them, *Befunde* is still wrongly predicted with utmost certainty. This opposition of both interpretability methods is observed for *Comprehensiveness* (Page 16), where an the inverse case occurs of IG assigning highest importance to a token, that receives negative attribution from SHAP.

***Mix* scores worst with SHAP.** With explanations from both interpretability methods, Mix is the worst-scored label when evaluated on *Sufficiency*. For a comparison of both methods, the below sample is given.

$$f(x)_{Mix} = 0.94, x = \texttt{"- Kostaufbau nach Ernährungskonsil"}.$$

Evaluating SHAP's explanation leads to the final mean *Sufficiency* score:

$$Suff = f(x)_{Mix} - f(r_{Mix})_{Mix} = 0.58.$$

In the sixth step the chosen tokens are: $r_{60} = [\texttt{Kost}_{0.41}, \texttt{ ,Ernährung}_{0.36}, \texttt{ skon}_{0.05}, \texttt{ il}_{0.1}]$, with the score $f(r_{60})_{Mix} = 0.03$. Those tokens are not *sufficient* for the model to predict *Mix*, instead *Medikation* is predicted $f(r_{60})_{Med} = 0.54$. However, by inclusion of one more token `"-"`, the model chooses *Mix* with $f(r_{70})_{Mix} = 0.70$, a 67 point increase in probability for the gold label.

***Mix* scores worst with IG.** When comparing IG's explanation, there is no such improvement in the *Sufficiency* scores along the steps. Quite to the contrary, all ten steps exhibit a score of $Suff >= 0.92$, meaning the chosen set of tokens r_{10-100} are in neither step *sufficient* to predict *Mix*. In fact, even all positively attributed tokens, constituting to 75% of the sequence, included in the last step as $r_{100} = [\texttt{Kost}_{0.41}, \texttt{ aufbau}_{0.03} \texttt{ ,Ernährung}_{0.36}, \texttt{ skon}_{0.05}, \texttt{ s}_{0.2} \texttt{ il}_{0.1}]$, *Befunde* is still the false positive with the probability $f(r_{100})_{Bfd} = 0.94$, in line with the probability *Mix*, when scoring the unmodified sequence $f(x)_{Mix}$. Also, the set of tokens r_{100} does not include the same token `"-"`, that causes the *Sufficiency* score for SHAP to improve substantially, because IG assigns a negative score to it.

With both SHAP's and IG's explanations evaluated on *ferret*'s *Faithfulness* through the sub-metrics *Comprehensiveness* and *Sufficiency*, the next chapter discusses these results.

6 Discussion

The following sections discuss and analyse the results from the previous chapter with regard to the best and worst performing label classes as well as the discrepancy between IG and SHAP in their evaluation against *ferret*'s *Faithfulness*.

6.1 Label-specific Analysis

There is no direct correlation between the labels' performances for *ferret*'s *Faithfulness* and their scores on the Sequence Classification task (Section 5.1). Nevertheless, insights can be gained on model confidence or insecurity towards classes by considering the properties of the data, that the model is trained on. Hence, through this analysis, the performances can be reasoned for in both the given task and *ferret*'s *Faithfulness*, even though they may diverge.

Anrede

Anrede scores best for *Comprehensiveness* with SHAP's explanations, whilst also having perfect scores on the classification task. The reason behind its top performance lies in its data properties. Anrede's sequences follow a strict pattern in structure and content, by which one one of them is approximately in 60% alignment with any other of its sequences [1]. This causes the model to easily recognize and hence, classify data of *Anrede*. Its score for *Comprehensiveness* with SHAP's explanations translates this, where by deletion of 10% of the highest contributors, *Anrede* loses all probability to *Abschluss*. This immediate loss of probability may on the one hand point to the label's top precision, through which the minimally modified sequence is regarded as a true negative. On the other hand, another reason besides the data homogeneity, also constituting to this may be *Anrede*'s small size in training data (Graph 1). Though the sequence is to the majority extent kept, the model still loses all confidence. Therefore, the *abrupt* decision of the model may be reinforced, due to an insecurity towards this minority class. The above state possible reasons

1 This statistic is obtained through a heuristic approach measuring the amount of intersecting tokens of a randomly chosen sequence with any other sequence. We run this three times, each with differently chosen sequences.

of why the probability for the gold label drops suddenly. However, it does not give insight into why the decision falls *confidently* on *Abschluss*.

Abschluss

Abschluss, contrary to *Anrede*, is a big label class, with the highest amount of sequences in the training data distribution (Graph 1). Though its token size, for comprising of small sequences (e.g.: `"I-PER / I."`), does not amount to those of other strongly represented classes (Graph 3), the model is trained most on this label. The model therefore, is more confident when considering *Abschluss* during predictions. *Ferret*'s *Comprehensiveness* again grasps this, in that a sequence deprived of the top 10% contributors from SHAP's explanation, does not lead to a total loss of probability for the gold label, like it is the case for *Anrede*. On the account of IG even, the same sequence with all positive contributing tokens truncated, scores higher for *Abschluss* than the minimally altered sequence in the first step through SHAP. Though IG's conspicuous explanations finds mention later in 6.2, both methods show the model rather strongly considers *Abschluss* throughout the steps of *Comprehensiveness*, in both cases of it being the gold label as well as a false positive for the sequence of *Anrede*.

Zusammenfassung

As mentioned in the above paragraph, the data of other classes surpass that of *Abschluss* in token size, such as *Zusammenfassung*. Its token amount is more than the ten-fold of that of *Mix*. Additionally, *Zusammenfassung* is 20 times richer in vocabulary than *Anrede*, putting it with a $1,000$ count difference right after the top candidate *Befunde*. With these properties, it is besides *Abschluss* a strongly represented label, through which the model sees the most amount and variety of tokens during training. *Sufficiency* reflects this. With SHAP's explanation for a respective sequence, a single token `"wurde"` is *sufficient* for the model to predict the gold label without doubt (Page 18). We note here, that the strong representation of label classes does not take away from the legitimacy of *ferret*'s *Sufficiency*, since it might be argued that for such edge cases, *any* token may be sufficient for the model, resulting in a lack of evaluation of the interpretability method against *Faithfulness*. What speaks against this is the fact, that even in such imbalanced classes, the choice of the *right* tokens still matters, shown by the scoring discrepancy between SHAP and IG for the same sample in Section 5.2.

Mix

While *Zusammenfassung*, *Befunde* and *Abschluss* are majority label classes, *Anrede* and *Mix* comprise the least amount of training data. The latter label, as mentioned above, is smaller in vocabulary than *Zusammenfassung* in absolute value. However, *Mix* contains the most amount of vocabulary relative to its token size. The contrary of these properties, namely scarceness in vocabulary together with richness in data (either token-wise or sequence-wise), poses a factor of model security towards such a class, as suggested by above analyses. Subsequently it might be argued, that *Mix*'s small data composition with high vocabulary proportion, results in model insecurity towards it. *Sufficiency* exhibits this by chosen tokens being substantially less *sufficient* for the model to be confident about, and thus predict *Mix*. With SHAP's explanation, only after including more than 60% of the whole sequence, the model classifies it correctly, whereas with IG's explanation, a coverage of all positively attributed tokens of the sequence (75%), doesn't guide the model in the right direction.

The above four labels either perform best or worst for *ferret*'s *Comprehensiveness* and *Sufficiency* through SHAP's explanations, always outperforming IG. This discrepancy is further analysed in the next section.

6.2 Method-specific Analysis

SHAP's explanations perform better for *ferret*'s *Comprehensiveness* and *Sufficiency* in all labels than those of IG (table 2, 3). With these results disclosed in the previous chapter 5 and analysed in the above section 6.1, two conspicuousness observations are made on behalf of one method. *Ferret*'s IG method for the former, assigns to more tokens negative attribution scores than SHAP does in all four sequences. Secondly, it - in three of those cases - judges a token to have the opposite contribution of what SHAP assigns to it. Both these cases for IG, negatively influence its performance for the evaluation against *ferret*'s *Faithfulness*. Since these observations are significant, they are closer regarded in the following.

Integrated Gradients assigns more negative contributions. For the former observation, we conduct an additional study, to test whether the inclusion of negatively attributed tokens might improve the evaluation of IG's explanations. The motivation behind this idea is the potential

those tokens could represent in a newly structured way, either to themselves or their surrounding tokens. We approach this by including the negative contributing tokens in each step of computing *Comprehensiveness* and *Sufficiency* by taking the absolute value of each token, hence with each step selecting the $k\%$ top positive and negative candidates. We test this on each the best- and worst-performing labels of *Comprehensiveness* and *Sufficiency* for both methods. This implementation we find, does not have any noteworthy effect on the mean scores of SHAP's two best performing labels, whilst it significantly harms the mean scores of the method's worst-scorers *Abschluss* and *Mix*, yielding $Comp_{Abs}^{SHAP} = 0.59$ and $Suff_{Mix}^{SHAP} = 0.67$ each. IG however, benefits off of the inclusion of its negatively attributed tokens in three out of four cases of the evaluation. For its worst-scorers *Anamnese* and *Mix*, the increase in score happens gradually throughout the steps, with a final mean score of $Comp_{Anm}^{IG} = 0.46$ to a prior $Comp_{Anm}^{IG} = 0.0$, as well as a $Suff_{Mix}^{IG} = 0.66$ to a prior $Suff_{Mix}^{IG} = 0.93$. Moreover, its best candidate *EchoBefunde* benefits substantially off of two negative tokens added at the beginning, with an end score of $Suff_{Ech}^{IG} = 0.08$ to a prior $Suff_{Ech}^{IG} = 0.34$. Consequently, this shows that *ferret*'s IG suffers from the exclusion of its negatively judged tokens when evaluated against *Comprehensiveness* and *Sufficiency*, thus confirming its misjudgement of tokens, that are actually contributing positively to the prediction of the correct class.

Integrated Gradients assigns contrasting contributions. The latter observation is about IG assigning attributions, which are from the opposite side of the spectrum to SHAP's decisions. To reiterate, `"ER"`, `"wurde"` and `"-"` are those tokens, where SHAP attributes the former two as positives and the last as a negative, while the contrary is true for Integrated Gradients (Pages 17, 19, 20). The methods disagreeing on those however, is only conspicuous, because they turn out to contribute significantly to the prediction of the gold label. Subsequently, the methods are evaluated very differently on the same sample, with the strongest case for *Sufficiency* on the majority class *Zusammenfassung* (Page 19). For IG's disregard of tokens, that are substantial for the correct classification, we suspect a problem with regard to its baseline.

The authors of IG state that features of the input sharing the value of the baseline, are guaranteed to have zero attribution [Sundararajan and Taly, 2018]. [Sturmfels et al., 2020] reference this property of IG by referring to it as "Blindness towards the baseline". They probe this by changing the color of the baseline for IG, which then generates different explanations for the same input on an Image Classification task. Therefore, IG proves to be "blind" to the color of the baseline, such that for a black image baseline, its explanations will not highlight black-coloured regions of the input.

Similarly, for our case we suggest, that IG may be "blind" to those tokens of a sequence, that score high for a majority class. For *ferret*'s IG we find, that the baseline sequence is built on a combination of special tokens from the BERT tokenizer, namely `"[CLS][PAD][SEP]"`, with its length set to the length of the explained sequence. Because the model does not see such a sequence during training, we suggest the baseline might score higher for majority label classes. We compare this and refer to the unintelligible text generated during *Sufficiency* (Page 18), which the model neither sees, but still scores unanimously for the majority class *Zusammenfassung*[2]. Therefore, IG's baseline likely leaning towards highly represented classes, may well explain the method's disregard for tokens, that score high for such classes. We obtain the following scores when classifying each token individually for the gold labels of the respective sequences (Pages 19, 17). $f(\texttt{"wurde"})_{Zsf=1.0}$ is disregarded while $f(\texttt{"Röntgen"})_{Zsf=0.0}$ considered and $f(\texttt{"P"})_{Abs=0.25}$ is left out while $f(\texttt{"ER"})_{Abs=0.0}$ is regarded. In both cases for the majority classes *Zusammenfassung* and *Abschluss*, IG chooses the lowest scoring tokens as highest contributing, whilst discrediting those that score higher individually and through SHAP's explanations are proven to contribute majorly (Pages 18, 16).

We note however, that the above reasoning to explain IG's tendency to disregard important tokens, come with a couple constraints. For one, our heuristic approach of scoring a sequence comparable to the baseline, cannot guarantee that majority classes are actually favoured in *ferret*'s version. Secondly, IG, for a sequence of the minority class *Mix*, also misjudges its high contributing token `"-"` (Page 20). If, as suggested, the contrast of the method stems from the bias of the baseline towards majority classes, and further supposed that is the only reason, then the high contributing token for the weakly represented *Mix*, should not lie in the "blind spot" of the baseline, thus receive attention for straying from the lean of the baseline. This is, as shown not the case (Page 20). Finally, since besides IG, Gradients also inherits this property due to its likewise architecture, the "blindness" should also apply to it. For *ferret*'s *Gradients*, we find this may not hold true. Concretely, the sample when explained (Page 19), has the attribution score $\mathit{Gradient}_{0.09}$ for the token `"wurde"`, thus judged to contribute positively for the majority class.

In view of the above discussions on the conspicuous observations for *ferret*'s IG, we opened an issue on GitHub[3] for the authors to take note on this.

---

2 Comparably scoring the string version of this baseline, we obtain results majorly in favour of big classes: $f(b)_{Zsf} = 0.73$, $f(b)_{Anm} = 0.21$, $f(b)_{Bfd} = 0.04$, $f(b)_{Ech} = 0.01$, $f(b)_{Abs} = 0.01$, where $b =$ `"[CLS] [PAD] [SEP]"`.

3 `https://github.com/g8a9/ferret/issues/21`

We additionally note, that *ferret*'s deployment of both *Comprehensiveness* and *Sufficiency* is an ideal way of evaluating a method against *Faithfulness* across the input space. The reason being, that minority classes tend to perform better for *Comprehensiveness*, while for *Sufficiency*, majority classes score higher (Section 6.1). In fact, the contrast in performance of a label class in those sub-metrics is a direct sign of model-bias, in our case *Mix* scoring best with IG (second best with SHAP) for *Comprehensiveness*, while performing worst with both methods for *Sufficiency*. Through exposure of bias, a method is not incorrectly deemed less *faithful* for having differences in its scoring across the input space or contrasting scores for a label. Rather, a method that is performing better for both *Comprehensiveness* and *Sufficiency* than another one, can be deemed "sufficiently" more *faithful*.

Through discussions on label- and method-specific aspects of the *Faithfulness* evaluation of SHAP and IG, we gain important insights on the evaluations of IG and SHAP and more generally, a reason for the former method's handling of concrete inputs. In the following chapter Conclusion, we summarize our work.

7 Conclusion

We conclude our work, by summarizing our findings with regard to our objective of finding a method that is more *faithful* than the other.

With regards to the *Faithfulness* evaluation we find, that *ferret*'s SHAP performs overall and significantly better than IG. From this we posit that *ferret*'s SHAP is a "sufficiently" *faithful* interpretation method on our outputs of the Sequence Classification task, while *ferret*'s IG is "sufficiently" *unfaithful* in the same regard. More concretely, we gain insight, that this stems from the latter method assigning attributions to tokens, that don't uphold once scored individually or in their respective context in the sequence, as shown per SHAP's explanations. On the potential reasons for this, we find research revealing that any value of the input corresponding to that of the baseline, will not be highlighted by IG. This we suggest, provides a strong case for why especially those inputs leaning towards majority label classes are disregarded by IG. However, we also underline the counter-case of a minority label class targeted in the same way, for which this we suggest should not hold true, if solely the baseline issue is the underlying reason.

On behalf of the Sequence Classification task, we find, that certain classes outperform others, due to their data properties, such that majority and minority classes are distinguished by sequence/token size and vocabulary proportion. Through this we observe, their performances, though not correlated, reflected in the sub-metrics *Comprehensiveness* and *Sufficiency*. With regards to both we find, that contrasting performances of a class in them, points at model bias caused by imbalanced distribution of data. Additionally, the step-wise approach, we find is an insightful way of assessing the grading ability of a feature attribution method, thus also providing a more in-depth comparison of different methods.

A Appendix

Method	Input Feature													
	Per	ip	here	Ö	dem	e	:	moder	ate	US	-	Ö.1	dem.1	e.1
Partition SHAP	0.14	0.02	-0.03	0.13	0.03	-0.04	0.19	0.11	0.01	-0.03	-0.01	-0.00	0.04	0.22
Integrated Gradients	-0.01	-0.01	-0.01	0.25	-0.16	-0.11	0.07	0.13	0.01	0.03	0.02	0.03	-0.07	-0.02

Table 4: Sample explanation by *ferret* from a sequence of label *KUBefunde*

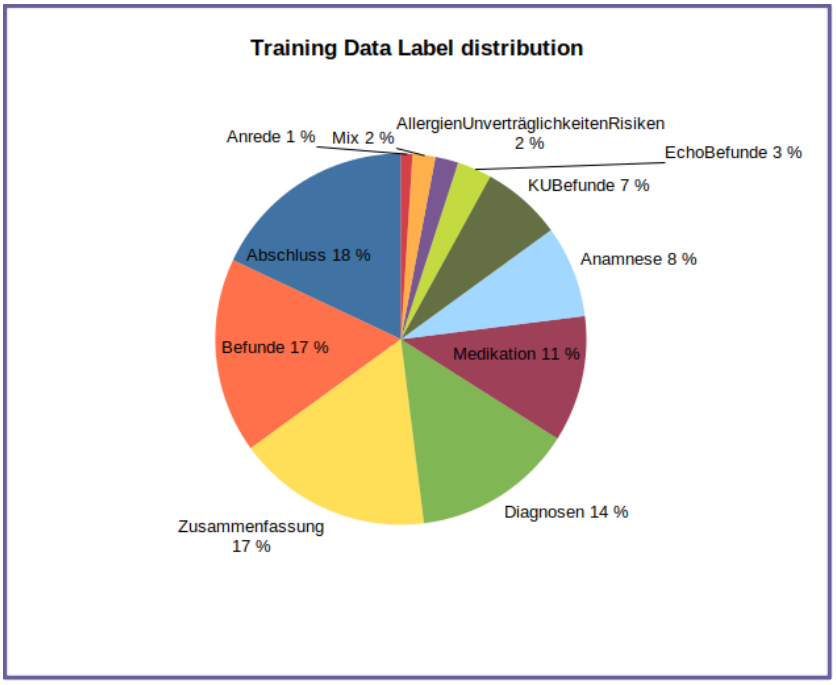

Figure 1: Data Distribution as Samples per Label in Training Data

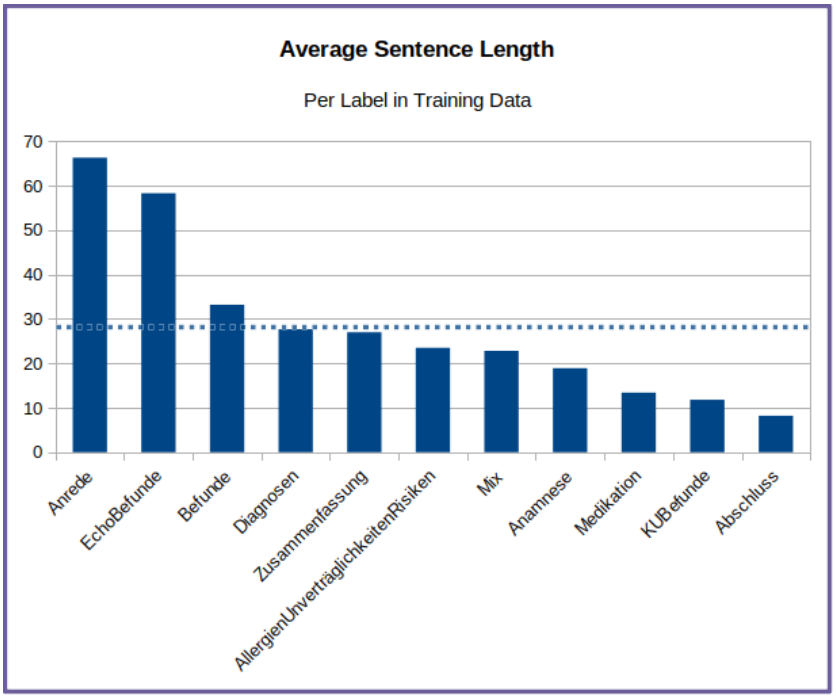

Figure 2: Average Sentence Length per Label in Training Data

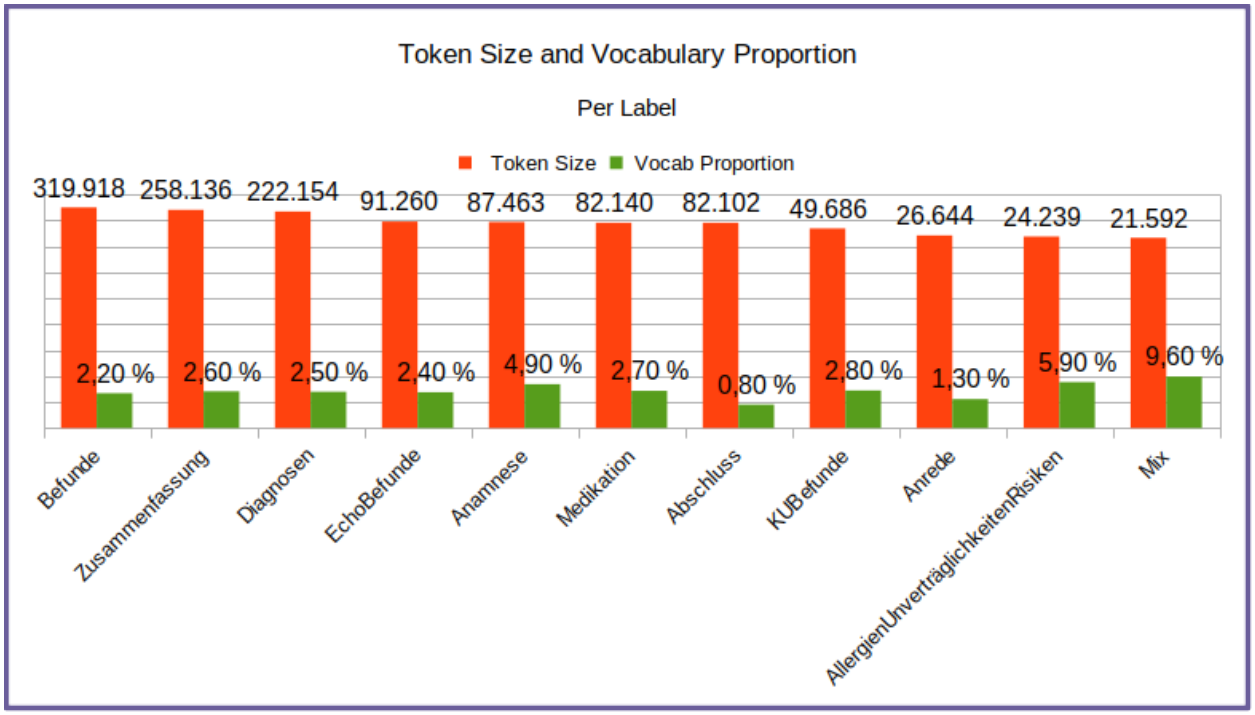

Figure 3: Token Size and Vocabulary Proportion per Label in Training Data

Bibliography

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. A survey of the state of explainable AI for natural language processing. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 447–459, Suzhou, China, December 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.aacl-main.46.

Giuseppe Attanasio, Eliana Pastor, Chiara Di Bonaventura, and Debora Nozza. ferret: a framework for benchmarking explainers on transformers. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, May 2023.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017. URL https://arxiv.org/abs/1703.01365.

Ashish Bora, Siva Balasubramanian, Boris Babenko, Sunny Virmani, Subhashini Venugopalan, Akinori Mitani, Guilherme de Oliveira Marinho, Jorge Cuadros, Paisan Ruamviboonsuk, Greg S Corrado, Lily Peng, Dale R Webster, Avinash V Varadarajan, Naama Hammel, Yun Liu, and Pinal Bavishi. Predicting the risk of developing diabetic retinopathy using deep learning. *The Lancet Digital Health*, 3(1):e10–e19, jan 2021. doi: 10.1016/s2589-7500(20)30250-8. URL https://doi.org/10.1016%2Fs2589-7500%2820%2930250-8.

Magnus D. Char DS, Shah NH. Implementing machine learning in health care - addressing ethical challenges. *N Engl J Med*, Mar 2018. doi: 10.1056/NEJMp1714229.

Laurie Lovett Novak, Regina G Russell, Kim Garvey, Mehool Patel, Kelly Jean Thomas Craig, Jane Snowdon, and Bonnie Miller. Clinical use of artificial intelligence requires AI-capable organizations. *JAMIA Open*, 6(2), 05 2023. ISSN 2574-2531. doi: 10.1093/jamiaopen/ooad028. URL https://doi.org/10.1093/jamiaopen/ooad028. ooad028.

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness?, 2020. URL https://arxiv.org/abs/2004.03685.

L.A. Harrington, M.D. Morley, A. Ščedrov, and S.G. Simpson. *Harvey Friedman's Research on the Foundations of Mathematics*. ISSN. Elsevier Science, 1985. ISBN 9780080960401. URL https://books.google.co.il/books?id=2plPRR4LDxIC.

Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation, 2019.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.

Wojciech Samek, Alexander Binder, Gregoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28:2660–2673, 11 2017. doi: 10.1109/TNNLS.2016.2599820.

Sofia Serrano and Noah A. Smith. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1282. URL https://aclanthology.org/P19-1282.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.408. URL https://aclanthology.org/2020.acl-main.408.

Mukund Sundararajan and Ankur Taly. A note about: Local explanation methods for deep neural networks lack sensitivity to parameter values, 2018.

Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. *Distill*, 2020. doi: 10.23915/distill.00022. https://distill.pub/2020/attribution-baselines.

Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 17(4):319–330, 2001. doi: https://doi.org/10.1002/asmb.446. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/asmb.446`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

Phillip Richter-Pechanski, Philipp Wiesenbach, Dominic Schwab, Christina Kiriakou, Mingyang He, Michael Allers, Anna Tiefenbacher, Nicola Kunz, Anna Martynova, Noemie Spiller, Julian Mierisch, Florian Borchert, Charlotte Schwind, Norbert Frey, Christoph Dieterich, and Nicolas Geis. A distributable german clinical corpus containing cardiovascular clinical routine doctor's letters. *Scientific Data*, 10, 04 2023. doi: 10.1038/s41597-023-02128-9.