On the Impact of Temporal Concept Drift on Model Explanations

Zhixue Zhao **George Chrysostomou** **Kalina Bontcheva** **Nikolaos Aletras**
Department of Computer Science, University of Sheffield
United Kingdom
{zhixue.zhao, gchrysostomou1, k.bontcheva, n.aletras}@sheffield.ac.uk

Abstract

Explanation faithfulness of model predictions in natural language processing is typically evaluated on held-out data from the same temporal distribution as the training data (i.e. synchronous settings). While model performance often deteriorates due to temporal variation (i.e. temporal concept drift), it is currently unknown how explanation faithfulness is impacted when the time span of the target data is different from the data used to train the model (i.e. asynchronous settings). For this purpose, we examine the impact of temporal variation on model explanations extracted by eight feature attribution methods and three select-then-predict models across six text classification tasks. Our experiments show that (i) faithfulness is not consistent under temporal variations across feature attribution methods (e.g. it decreases or increases depending on the method), with an attention-based method demonstrating the most robust faithfulness scores across datasets; and (ii) select-then-predict models are mostly robust in asynchronous settings with only small degradation in predictive performance. Finally, feature attribution methods show conflicting behavior when used in FRESH (i.e. a select-and-predict model) and for measuring sufficiency/comprehensiveness (i.e. as post-hoc methods), suggesting that we need more robust metrics to evaluate post-hoc explanation faithfulness.[1]

1 Introduction

One way of improving the transparency of deep learning models in natural language processing (NLP) is by extracting explanations that justify model predictions (Lipton, 2018; Guidotti et al., 2018). An explanation (i.e. rationale) consists of a subset of the input and is considered faithful when it accurately shows the reasoning behind a model's prediction (Zaidan et al., 2007; Ribeiro et al., 2016a; DeYoung et al., 2019; Jacovi and Goldberg, 2020). For example, removing a faithful rationale from the input should result into a prediction change. Two widely used methods for extracting rationales are (i) feature attribution methods that produce a distribution over the input tokens, indicating their contribution (i.e. importance) to the model's prediction (Ribeiro et al., 2016b; Wiegreffe and Pinter, 2019); and (ii) select-then-predict models that consist of two components, a rationale extractor and a predictor. The rationale extractor extracts rationales, and the predictor is trained on extracted rationales so that its predictions are inherently faithful (Lei et al., 2016; Jain et al., 2020).

Previous work has focused on evaluating explanation faithfulness in *synchronous* settings where the training and testing data come from the same temporal distribution (Serrano and Smith, 2019a; Jain and Wallace, 2019; Atanasova et al., 2020; Guerreiro and Martins, 2021), or out-of-domain settings (Chrysostomou and Aletras, 2022a) where the training and testing data come from a different domain regardless of temporal drifts in the testing data. However, human languages evolve (Weinreich et al., 1968; Kim et al., 2014; Carrier, 2019) as manifested by novel usages developed for existing words (e.g. *mouse* is a mammal or a computer accessory) and new words and topics (e.g. *covidiot* during the COVID-19 pandemic) that appear over time. Language evolution leads to temporal concept drifts and a diachronic degradation of model performance in many NLP tasks when these are evaluated in *asynchronous* settings, i.e. training and testing data come from different time periods (Jaidka et al., 2018; Agarwal and Nenkova, 2021; Lazaridou et al., 2021; Søgaard et al., 2021; Chalkidis and Søgaard, 2022).

In this paper, for the first time, we extensively analyze the impact of temporal concept drift on model explanations. We evaluate the faithfulness of

[1] Code for replicating the experiments in this study: https://github.com/casszhao/temporal-drift-on-explanation

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 4068 - 4083
December 7-11, 2022

rationales extracted using eight feature attribution approaches and three select-then-predict models over six text classification tasks with chronological data splits. Our contributions are as follows:

- We find that faithfulness is not consistent under temporal concept drift for rationales extracted with feature attribution methods (e.g. it decreases or increases depending on the method), with an attention-based method demonstrating the most robust faithfulness scores across datasets;
- We empirically show that select-then-predict models can be used in asynchronous settings when it achieves comparable performance to the full-text model;
- We demonstrate that sufficiency is not trustworthy evaluation metrics for explanation faithfulness, regardless of a synchronous or an asynchronous setting.

2 Related Work

2.1 Temporal Concept Drift in NLP

Temporal model deterioration describes the *true difference in system performance* when a system is evaluated on chronologically newer data (Jaidka et al., 2018; Gorman and Bedrick, 2019). This has been linked to changes in the data distribution, also known as *concept drift* in early studies (Schlimmer and Granger, 1986; Widmer and Kubat, 1993). Previous work has demonstrated the impact of temporal concept drift on model performance by assessing the *temporal generalization* (Lazaridou et al., 2021; Søgaard et al., 2021; Agarwal and Nenkova, 2021; Röttger and Pierrehumbert, 2021). Søgaard et al. (2021) has studied several factors that affect the true difference in system performance such as temporal drift, variations in text length and adversarial data distributions. They found that temporal variation is the most important factor for performance degradation and suggest including chronological data splits in model evaluation. Chalkidis and Søgaard (2022) also noted that evaluating on random splits with the same temporal distribution as the training data consistently over-estimates model performance at test time in multi-label classification problems.

Previous work on mitigating temporal concept drift includes automatically identifying semantic drift of words over time (Tsakalidis et al., 2019; Giulianelli et al., 2020; Rosin and Radinsky, 2022; Montariol et al., 2021). Efforts have also been made to mitigate the impact of temporal concept drift on model prediction performance (Lukes and Søgaard, 2018; Röttger and Pierrehumbert, 2021; Loureiro et al., 2022; Chalkidis and Søgaard, 2022) and develop time-aware models (Dhingra et al., 2022; Rijhwani and Preotiuc-Pietro, 2020; Dhingra et al., 2021; Rosin and Radinsky, 2022). For example, both Röttger and Pierrehumbert (2021) and Loureiro et al. (2022) observed performance improvements when continue fine-tuning their models with chronologically newer data. While the impact of temporal concept drift on model performance has received particular attention, to the best of our knowledge, no previous work has examined its impact on model explanations.

2.2 Concept Drift and Model Explanations

Poerner et al. (2018) has compared the explanation quality between tasks that contain short and long textual context. More recently, Chrysostomou and Aletras (2022a) have studied model explanations in out-of-domain settings (i.e. under concept drift) using train and test data from different domains. Their results showed that the faithfulness of out-of-domain explanations unexpectedly increases, i.e. outperforming in-domain explanations' faithfulness. This is interesting given that performance degradation due to concept drift is often expected in domain adaptation (Schlimmer and Granger, 1986; Widmer and Kubat, 1993; Chan and Ng, 2006; Gama et al., 2014).

3 Extracting Explanations

We extract explanations using two standard approaches: (i) post-hoc methods; and (ii) select-then-predict models.

3.1 Post-hoc Explanation Methods

For post-hoc explanations, we fine-tune a BERT-base model on each task on the synchronous training set and extract explanations using post-hoc feature attribution methods for all synchronous and asynchronous testing sets. We use eight widely used feature attribution methods following Chrysostomou and Aletras (2021a,b).

- **Attention (α):** Token importance is computed using the corresponding normalized attention scores (Jain et al., 2020).

- **Scaled attention** ($\alpha\nabla\alpha$) Attention scores scaled by their corresponding gradients (Serrano and Smith, 2019a).
- **InputXGrad** ($x\nabla x$) Attributes importance by multiplying the input with its gradient computed with respect to the predicted class (Kindermans et al., 2016; Atanasova et al., 2020).
- **Integrated Gradients (IG)** Ranks input tokens by computing the integral of the gradients taken along a straight path from a baseline input (zero embedding vector) to the original input (Sundararajan et al., 2017).
- **GradientSHAP (Gsp)** A gradient-based method to compute SHapley Additive exPlanations (SHAP) values for assigning token importance (Lundberg and Lee, 2017). Gsp computes the gradient of outputs with respect to randomly selected points between the inputs and a baseline distribution.
- **LIME** Ranks input tokens by learning a linear surrogate model using data points randomly sampled locally around the prediction (Ribeiro et al., 2016b).
- **DeepLift (DL)** Computes token importance according to the difference between the activation of each neuron and a reference activation (i.e. zero embedding vector) (Shrikumar et al., 2017).
- **DeepLiftSHAP (DLsp)** Similar to Gsp, DLsp computes the expected value of attributions based on DL across all input-baseline pairs, considering a baseline distribution (Lundberg and Lee, 2017).

3.2 Select-then-predict Models

We also use three state-of-the-art select-then-predict models. Two are trained end-to-end (Bastings et al., 2019; Guerreiro and Martins, 2021) while the other one uses a feature attribution method as the rationale extractor (Jain et al., 2020) with a separate predictor component, trained on the extracted rationales.

- **HardKUMA:** Bastings et al. (2019) proposed a modified version of the end-to-end rationale extraction model introduced by Lei et al. (2016). Choosing rationales in a binary fashion by sampling from a Bernoulli distribution is replaced with a Kumaraswamy distribution (Kumaraswamy, 1980) to support continuous random variables. This way, the model is differentiable and easier to train.
- **SPECTRA:** HardKUMA provides stochastic rationales due to the marginalization over all possible rationales and the sampling process. Guerreiro and Martins (2021) proposed SPECTRA, a model that uses LP-SparseMAP (Niculae and Martins, 2020) to obtain a deterministic rationale extraction process. Niculae and Martins (2020) have experimented with three different factor graphs showing that XorAtMostOne outperforms the other two (i.e. Budget, AtMostOne2). We use SPECTRA with XorAtMostOne in our experiments. For HardKUMA and SPECTRA, we use a Bi-LSTM (Hochreiter and Schmidhuber, 1997) because it has been shown to outperform BERT-based models (Guerreiro and Martins, 2021).
- **FRESH:** Jain et al. (2020) proposed FRESH, a model that first extracts rationales from a trained model (e.g. using a feature attribution method) and subsequently trains a classifier on the extracted rationales. We extract the top 20% rationales using $\alpha\nabla\alpha$ that achieved the best performance in early experimentation. We also use BERT-base for the extraction and predictor components following Jain et al. (2020).

4 Experimental Setup

4.1 Tasks and Data

Tasks We evaluate all methods on three diverse text classification tasks including six different datasets: (1) topic classification; (2) misinformation detection; and (3) sentiment analysis:

- **AGNews:** Topic classification across four topics (Business, Sports, Science/Technology and World) from AG News (Del Corso et al., 2005);
- **X-FACT:** Factual correctness classification of short statements into five classes (Gupta and Srikumar, 2021): True, Mostly-True, Partly-True, Mostly-False and False;
- **FactCheck:** Binary classification of potential misinformation stories as truthful or misinformation (Jiang and Wilson, 2021);

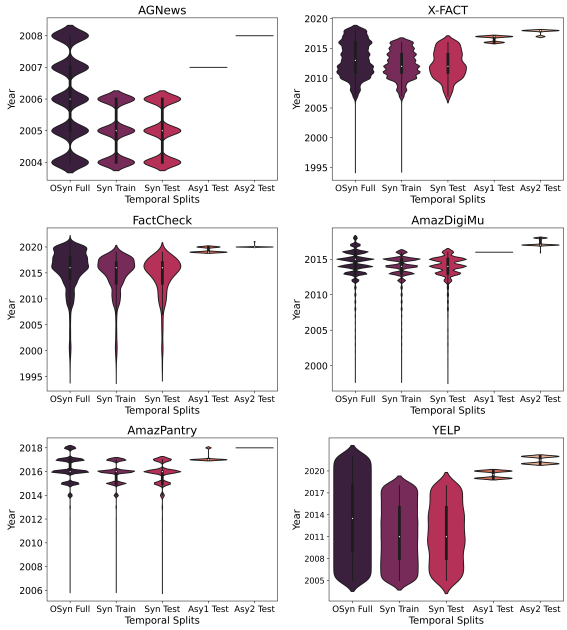

Figure 1: Density curves for time distribution across temporal splits and the original full size dataset for each task.

Task	#Classes	Splits	Start Date	End Date	Span (Days)	#Data
AGNews	4	Train	2004-08-18	2006-12-20	854	9358
		Syn Test	2004-08-18	2006-12-20	854	9358
		Asy1 Test	2007-01-30	2007-12-31	335	9358
		Asy2 Test	2008-01-01	2008-02-20	50	9358
X-FACT	6	Train	1995-04-01	2016-08-31	7823	7232
		Syn Test	2007-01-04	2016-08-31	3527	1204
		Asy1 Test	2016-08-31	2017-09-30	395	1205
		Asy2 Test	2017-09-30	2018-11-12	408	1204
FactCheck	2	Train	1995-09-25	2019-05-01	8619	7446
		Syn Test	1996-08-02	2019-05-01	8307	1241
		Asy1 Test	2019-05-02	2020-05-15	379	1368
		Asy2 Test	2020-05-15	2021-07-19	430	1368
AmazDigiMu	3	Train	1998-08-21	2016-05-07	6469	101774
		Syn Test	1998-12-20	2016-05-07	6351	16963
		Asy1 Test	2016-05-07	2016-12-30	237	16962
		Asy2 Test	2016-12-30	2018-09-26	635	16962
AmazPantry	3	Train	2006-04-28	2017-07-30	4111	82566
		Syn Test	2006-12-22	2017-07-30	3873	13762
		Asy1 Test	2017-07-30	2018-01-21	175	13761
		Asy2 Test	2018-01-21	2018-10-04	256	13761
Yelp	5	Train	2005-02-16	2018-12-31	5066	8540
		Syn Test	2005-02-16	2018-12-24	5059	1708
		Asy1 Test	2019-01-01	2020-12-31	730	1708
		Asy2 Test	2021-01-01	2022-01-19	383	1708

Table 1: Data statistics and the temporal splits for each task.

- **Amazon Reviews:** We predict the sentiment (negative, neutral, positive) of Amazon product reviews from digital music (**AmazDigiMu**) and pantry (**AmazPantry**) as Ni et al. (2019);
- Yelp: Multi-class sentiment classification (positive, negative) following Zhang et al. (2015).

Data Splits To simulate temporal concept drifts, we create different chronological splits according to the time-stamps of the data points in each dataset. We split each dataset into a training set and three different test sets. The time spans of the three test sets follow a chronological order without any overlapping. The test set with the earliest time span (*Syn*) has the exact same time span as the training data (i.e. a synchronous setting). The other two splits denoted as *Asy1* and *Asy2* that are chronologically newer correspond to asynchronous settings. Figure 1 shows the temporal distribution of each data split compared to the original data. Table 1 summarizes the key statistics for each split. More details for the data and tasks can be found in the Appendix A. We also provide results of all models on the original (synchronous) test set (*OSyn*).

4.2 Evaluation

For each task, we train a model on the training set and then evaluate post-hoc explanations and select-then-predict performance on our three chronological splits, namely *Syn*, *Asy1* and *Asy2*.

Post-hoc Explanations We evaluate the faithfulness of post-hoc explanations using two popular metrics (DeYoung et al., 2019; Carton et al., 2020):

- **Normalized Sufficiency** quantifies how sufficient a rationale is for making the same prediction $p(\hat{y}|\mathcal{R})$ to the prediction of the full text model $p(\hat{y}|\mathbf{x})$. We use the normalized version to allow a fairer comparison across models and tasks:

$$\text{Suff}(\mathbf{x}, \hat{y}, \mathcal{R}) = 1 - max(0, p(\hat{y}|\mathbf{x}) - p(\hat{y}|\mathcal{R}))$$
$$\text{NormSuff}(\mathbf{x}, \hat{y}, \mathcal{R}) = \frac{\text{Suff}(\mathbf{x}, \hat{y}, \mathcal{R}) - \text{Suff}(\mathbf{x}, \hat{y}, 0)}{1 - \text{Suff}(\mathbf{x}, \hat{y}, 0)} \quad (1)$$

- **Normalized Comprehensiveness** assesses how much information the rationale holds, measuring changes in predictions when masking the rationale $p(\hat{y}|\mathbf{x}_{\setminus\mathcal{R}})$. Similar to sufficiency, we use the normalized version:

$$\text{Comp}(\mathbf{x}, \hat{y}, \mathcal{R}) = max(0, p(\hat{y}|\mathbf{x}) - p(\hat{y}|\mathbf{x}_{\setminus\mathcal{R}}))$$
$$\text{NormComp}(\mathbf{x}, \hat{y}, \mathcal{R}) = \frac{\text{Comp}(\mathbf{x}, \hat{y}, \mathcal{R})}{1 - \text{Suff}(\mathbf{x}, \hat{y}, 0)} \quad (2)$$

Further, we evaluate explanations of different lengths (top 2%, 10%, 20% and 50% of tokens extracted) and report the "Area Over the Perturbation

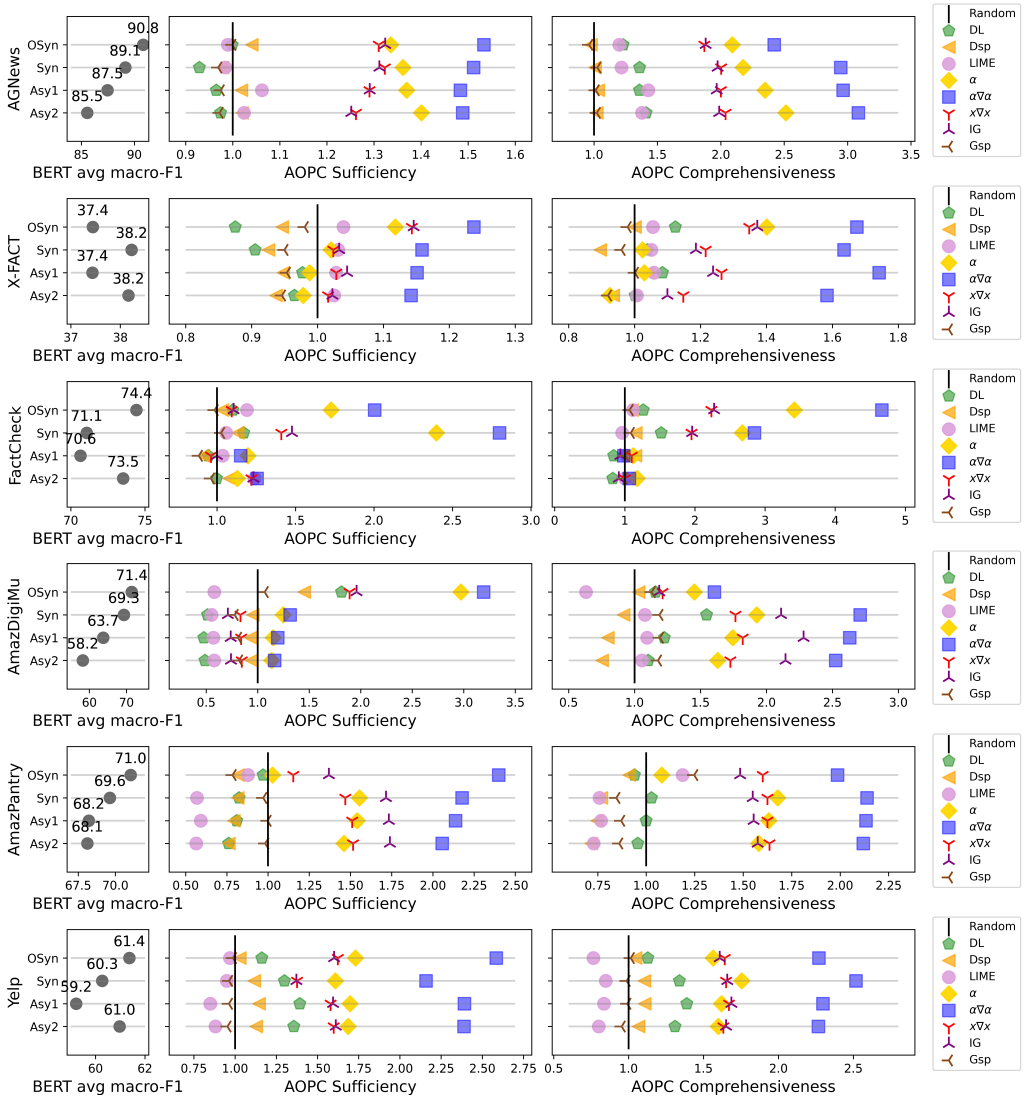

Figure 2: Post-hoc AOPC normalized sufficiency and comprehensiveness (higher is better) of the original test set (OSyn), our Syn and Asy splits, across feature attribution approaches and a random baseline.

Curve" (AOPC) of Normalized Sufficiency and Normalized Comprehensiveness following DeYoung et al. (2019).

Following Chrysostomou and Aletras (2022a), a baseline random attribution (i.e. randomly assigning importance scores) is used as a yardstick to allow a comparison across chronological splits and tasks. We use the ratio between the comprehensiveness or sufficiency score of a feature attribution and the score of the random baseline to compute its final faithfulness score. Faithfulness scores under 1.0 indicate that the rationale for a particular feature attribution is less faithful than just randomly selecting input tokens as a rationale.

We avoid using metrics such as RemOve And Retrain (ROAR) due to their demanding computation requirements (Hooker et al., 2019). We also omit the use of other popular metrics such as Word Relevance (Arras et al., 2019) and Fraction of Tokens (Serrano and Smith, 2019b) as they are similar to comprehensiveness and sufficiency.

Select-then-predict Select-then-predict classifiers are trained only on rationales and discard the rest of the input, hence they are inherently faithful. As such, one way to check how good their extracted explanations are, is to compare their predictive performance to the full-text trained model following Jain et al. (2020). A good rationale should achieve high predictive performance retention compared to the full-text model. We, therefore, compare the predictive performance of the three select-then-predict models with corresponding models trained on full-text: (1) FRESH against a BERT-base model; and (2) HardKUMA and SPECTRA against a Bi-LSTM

with the same number of layers, pre-trained embeddings and hidden dimensions.

5 Results

5.1 Post-hoc Explanation Faithfulness

We hypothesize that when the predictive performance of a model drops in asynchronous settings, the sufficiency and faithfulness scores that are based on predictive likelihood should also drop. Our hypothesis is based on the assumption that the lower the predictive performance, the lower the predictive likelihood for a well-calibrated model (Desai and Durrett, 2020).

Figure 2 shows the AOPC normalized sufficiency and comprehensiveness scores for each feature attribution method across temporal splits, with the corresponding model predictive performance along the left side. Full results can be found in Table 3 in the Appendix.

We first observe that certain feature attributions (e.g. $x\nabla x$, $\alpha\nabla\alpha$, IG) score above the random baseline in the majority of settings, suggesting that they remain faithful in asynchronous settings. The attention-based $\alpha\nabla\alpha$ in particular, outperforms not only the random baseline by a large margin in all settings, but also the rest of the feature attributions tested in the majority of cases. For example, across all temporal splits in Yelp, $\alpha\nabla\alpha$ scores higher than 2.16 in both sufficiency and comprehensiveness, i.e. compared to the random baseline. The second-best one, α, is again an attention-based attribution method and only scores 1.60x better. Two other feature attribution methods (LIME and Gsp) fail to score above the random baseline. On the other hand, certain methods such as DL, also fail to outperform the random baseline in general. This suggests that they cannot be trusted in asynchronous settings. For instance, Gsp fails to exceed the random baseline across all tasks for sufficiency scores on Asy splits while DL only scores slightly higher than random baseline (1.36 and 1.37 respectively).

Contrary to our initial hypothesis, *the faithfulness scores of all feature attribution methods do not necessarily fluctuate together with predictive performance, when comparing between synchronous and asynchronous settings*.

For example, in AGNews, AmazDigiMu and AmazPantry, predictive performance decreases along with chronological order. However, looking into sufficiency, only $x\nabla x$ in AGNews, $\alpha\nabla\alpha$ and α in both AmazDigiMu and AmazPantry con-

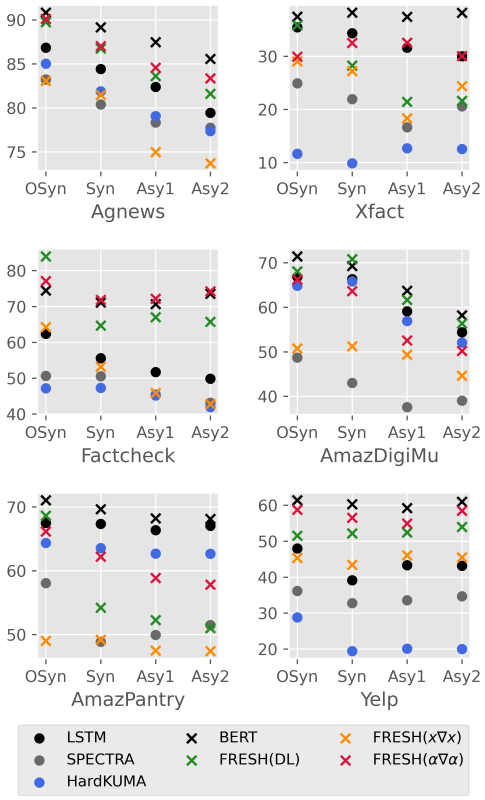

Figure 3: Averaged macro F1 performance (5 runs) of select-then-predict methods and models trained on full-text.

firm our initial hypothesis, i.e. the faithfulness decreases along with predictive performance. In the rest of the cases for these three tasks, we do not observe any pattern between faithfulness and the chronological order of the test data.

5.2 Select-then-predict Predictive Performance

Figure 3 shows the macro F1 scores (i.e. averaged over 5 runs with different random seeds) for the three select-then-predict models and their full-text trained counterparts. We compare full-text trained BERT against FRESH using the most faithful feature attribution ($\alpha\nabla\alpha$), one that is close to the average faithfulness of all methods ($x\nabla x$) and the least faithful one (DL). We also compare HardKUMA and SPECTRA against a full-text trained Bi-LSTM. For the full stack of results (including standard deviations), see Appendix D.

HardKUMA & SPECTRA As expected, the two models result in predictive performance drops compared to the full-text trained Bi-LSTM in the synchronous data splits (i.e. OSy and Syn). In asynchronous settings (i.e. Asy1 and Asy2), there

(a) FactCheck (Syn)

(b) FactCheck (Asy2)

(c) Yelp (Syn)

(d) Yelp (Asy2)

Figure 4: Wordclouds from synchronous (Syn) and asynchronous (Asy2) test sets for FactCheck and Yelp.

is no consistent pattern observed. In certain tasks, their predictive performance increases (e.g. SPECTRA in AmazPantry in both Asy1 and Asy2) while in other tasks decreases (e.g. for SPECTRA and HardKUMA in AGNews). For example, the performance of HardKUMA drops gradually from 85% for OSyn to 77% in Asy2. Similarly, the performance for SPECTRA drops from 50% to approximately 40% in FactCheck.

An interesting observation is that the predictive performance of HardKUMA and SPECTRA is comparable to the model trained on full-text in asynchronous settings in cases the two models have also achieved a comparable performance in synchronous settings. For example, HardKUMA exhibits comparable performance across all settings with the full-text trained Bi-LSTM in AmazDigiMu.

We therefore suggest, that HardKUMA and SPECTRA are reliable in asynchronous settings, when only their performance is comparable to the full-text model in synchronous settings.

FRESH We hypothesize that FRESH trained with rationales extracted from a faithful feature attribution method (i.e. its sufficiency is substantially higher relative to the random baseline), it should result into comparable predictive performance of the full-text trained model. In theory, these rationales should contain 'sufficient' information for a classifier to perform comparably to the full-text trained model.

We first observe that FRESH with $\alpha\nabla\alpha$ generally mirrors BERT's performance across all settings in most tasks, with the only exception in X-FACT (see Figure 3). We speculate that a possible reason for this, is the larger number of classes together with the small size of the dataset. This behavior also indicates that FRESH using the most faithful attention-based attribution method, $\alpha\nabla\alpha$ is not impacted by temporal drifts, more than its full-text trained counterpart. For example, the performance of FRESH($\alpha\nabla\alpha$) in X-FACT, FactCheck remains mostly stable across different test splits. In comparison, we do not observe the same mirroring behavior of FRESH train with less sufficient rationales from attribution methods such as DL and $x\nabla x$, across splits and tasks.

Comparing between faithfulness scores (sufficiency and comprehensiveness) and FRESH predictive performance, we identify a counter-intuitive pattern. In sharp contrast to our initial expectations, using rationales extracted from the lowest scoring feature attribution for sufficiency (i.e. DL), results in higher predictive performance for FRESH compared to the more sufficient rationales extracted with $x\nabla x$. For example, in AGNews, DL consistently scores below the random baseline for sufficiency in all settings, whilst $x\nabla x$ remains consistently more sufficient, scoring above the random baseline (see Figure 2). However, the performance of FRESH trained on rationales extracted with DL is directly comparable to using $\alpha\nabla\alpha$ for rationale extraction across all settings. Its performance is also closer to the full-text trained model. Using $x\nabla x$ rationales to train FRESH, it results into lower predictive performance compared to FRESH(DL). We further investigate these conflicting patterns in Section 7.

To summarize, *FRESH trained on rationales extracted by a robust and faithful feature attribution*

(i.e. $\alpha\nabla\alpha$) is reliable in asynchronous settings and not impacted by temporal drifts compared to the full-text model. FRESH trained with less faithful rationales, such as DL and $x\nabla x$, is not reliable across tasks.

6 Qualitative Analysis

We also conduct a qualitative analysis on the rationales extracted by $\alpha\nabla\alpha$, to find possible reasons that justify its stability and robustness when moving from synchronous to asynchronous settings (i.e. invariance to concept drift), when using FRESH. Figure 4 shows wordclouds (larger words appear more frequently in the rationales) for FactCheck (a, b) and Yelp (c, d), on the synchronous split Syn and the asynchronous split Asy2.

Starting with FactCheck, we observe that salient words change when moving to asynchronous settings. For example, in Syn, the extracted rationales contain words like "web", "site" and "published". In contrast, rationales from Asy2 contain words like "video", "social" and "post". These indicate a shift in how misinformation is spread across time (i.e. different types of media), which surprisingly is picked up by the rationales when moving to asynchronous settings. Similarly in Yelp, whilst the majority of most frequent words remains similar across chronological splits (e.g. "delicious", "nice", "amazing"), we still observe some concept drift that is again picked up by the rationales. For example, it appears that more recent restaurant reviews are concerned about the experience and appearance of the restaurant, as picked up in Asy2. This is highlighted by the fact that Asy2 contains frequently the words "experience" and "pretty" (to describe the place and food, we found several examples through a manual analysis that refer to these two concepts). On the other hand, we note that Syn does not contain words relevant to the experience and appearance of the place.

7 To Trust Sufficiency or Not?

The contradictory patterns observed between post-hoc explanations and FRESH (see Section 5), questions the efficacy of using sufficiency to measure faithfulness in asynchronous settings. Inspired by the explanation-game (Treviso and Martins, 2020), we use the classifier from FRESH as a layperson and measure its ability to generate the same predictions as the full-text trained model. Our hypothesis is that if a feature attribution produces highly sufficient rationales, the layperson should also have a high predictive performance (when using the full-text model's predictions as gold labels) and vice versa. If sufficiency is reliable as a metric, we therefore expect that the most sufficient rationales to be obtained using $\alpha\nabla\alpha$, followed by $x\nabla x$ and the least sufficient to be DL.

Figure 5 shows the performance of the layperson (i.e. FRESH), in predicting the original predictions of the full-text trained model (i.e. a higher score denotes a higher agreement). We first observe that $\alpha\nabla\alpha$ outperforms in most datasets both DL and $x\nabla x$. For example, in FactCheck, $\alpha\nabla\alpha$ shows almost perfect agreement (approximately 100%) with the full-text trained model in Syn and both Asy splits, highlighting the efficacy of this feature attribution in extracting faithful explanations.

Contrary to our expectations and similar to observations with FRESH, DL consistently outperforms $x\nabla x$, even outperforming $\alpha\nabla\alpha$ in certain cases. For example in Yelp across both asynchronous settings, using DL the layperson is able to reach the same predictions as the full-text model, in approximately 80% of the instances. In comparison, using $x\nabla x$, the layperson reaches the same predictions in only 55% of the instances.

Our findings suggest that *sufficiency, as a metric for measuring faithfulness, cannot be trusted in asynchronous settings and also raises concerns for synchronous settings.*

8 Conclusion

We conducted an extensive empirical study to shed light on the impact of temporal drift on model explanations in asynchronous settings, including post-hoc methods and select-then-predict models. We demonstrate that faithfulness is not consistent under temporal variations across feature attribution methods, while select-then-predict models are mostly robust with negligible drops in predictive performance. In the future, we plan to extend our study into more tasks and data from different languages. We also plan to explore whether instance specific feature attribution improves faithfulness in asynchronous settings (Chrysostomou and Aletras, 2022b).

Limitations

This study focuses only on experimenting with data in English. We would expect that the behavior of some methods might change due to linguistic id-

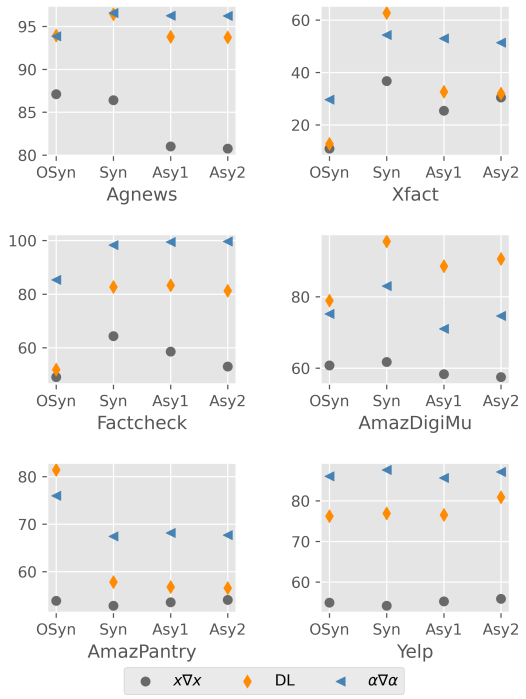

Figure 5: Averaged macro F1 performance of FRESH trained on $\alpha\nabla\alpha$, DL and $x\nabla x$ to predict the labels predicted by a full-text model (BERT). .

iosyncrasies across different languages. We believe that this is a very important direction for future work. Replicating our experiments requires access to GPUs.

Acknowledgements

ZZ, KB and NA are supported by EPSRC grant EP/V055712/1, part of the European Commission CHIST-ERA programme, call 2019 XAI: Explainable Machine Learning-based Artificial Intelligence. This project made use of time on Tier 2 HPC facility JADE2, funded by EPSRC (EP/T022205/1).

References

Oshin Agarwal and Ani Nenkova. 2021. Temporal effects on pre-trained models for language processing tasks. *arXiv preprint arXiv:2111.12790*.

Leila Arras, Ahmed Osman, Klaus-Robert Müller, and Wojciech Samek. 2019. Evaluating recurrent neural network explanations. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 113–126, Florence, Italy. Association for Computational Linguistics.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2963–2977, Florence, Italy. Association for Computational Linguistics.

Michael Carrier. 2019. Because internet: Understanding the new rules of language. *Training, Language and Culture*, 3(3):107–111.

Samuel Carton, Anirudh Rathore, and Chenhao Tan. 2020. Evaluating and characterizing human rationales. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9294–9307, Online. Association for Computational Linguistics.

Ilias Chalkidis and Anders Søgaard. 2022. Improved multi-label classification under temporal concept drift: Rethinking group-robust algorithms in a label-wise setting. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2441–2454, Dublin, Ireland. Association for Computational Linguistics.

Yee Seng Chan and Hwee Tou Ng. 2006. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, page 89–96, USA. Association for Computational Linguistics.

George Chrysostomou and Nikolaos Aletras. 2021a. Enjoy the salience: Towards better transformer-based faithful explanations with word salience. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8189–8200, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

George Chrysostomou and Nikolaos Aletras. 2021b. Improving the faithfulness of attention-based explanations with task-specific information for text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 477–488, Online. Association for Computational Linguistics.

George Chrysostomou and Nikolaos Aletras. 2022a. An empirical study on explanations in out-of-domain settings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6920–6938, Dublin, Ireland. Association for Computational Linguistics.

George Chrysostomou and Nikolaos Aletras. 2022b. Flexible instance-specific rationalization of nlp models. AAAI.

Gianna M. Del Corso, Antonio Gullí, and Francesco Romani. 2005. Ranking a stream of news. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, page 97–106, New York, NY, USA. Association for Computing Machinery.

Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. 2021. Time-aware language models as temporal knowledge bases. *arXiv preprint arXiv:2106.15110*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-Aware Language Models as Temporal Knowledge Bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

João Gama, Indrundefined Žliobaitundefined, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. *ACM Comput. Surv.*, 46(4).

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2786–2791, Florence, Italy. Association for Computational Linguistics.

Nuno M. Guerreiro and André F. T. Martins. 2021. SPECTRA: Sparse structured text rationalization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6534–6550, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42.

Ashim Gupta and Vivek Srikumar. 2021. X-fact: A new benchmark dataset for multilingual fact checking. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 675–682, Online. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. 2019. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Kokil Jaidka, Niyati Chhaya, and Lyle Ungar. 2018. Diachronic degradation of language models: Insights from social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 195–200, Melbourne, Australia. Association for Computational Linguistics.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. 2020. Learning to faithfully rationalize by construction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4459–4473.

Shan Jiang and Christo Wilson. 2021. Structurizing misinformation stories via rationalizing fact-checks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 617–631, Online. Association for Computational Linguistics.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. 2014. Temporal analysis of language through neural language models. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 61–65, Baltimore, MD, USA. Association for Computational Linguistics.

Pieter-Jan Kindermans, Kristof Schütt, Klaus-Robert Müller, and Sven Dähne. 2016. Investigating

the influence of noise and distractors on the interpretation of neural networks. *arXiv preprint arXiv:1611.07270*.

Ponnambalam Kumaraswamy. 1980. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2):79–88.

Angeliki Lazaridou, Adhi Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomas Kocisky, Sebastian Ruder, et al. 2021. Mind the gap: Assessing temporal generalization in neural language models. *Advances in Neural Information Processing Systems*, 34:29348–29363.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.

Zachary C Lipton. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-collados. 2022. TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260, Dublin, Ireland. Association for Computational Linguistics.

Jan Lukes and Anders Søgaard. 2018. Sentiment analysis under temporal shift. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 65–71, Brussels, Belgium. Association for Computational Linguistics.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Syrielle Montariol, Matej Martinc, and Lidia Pivovarova. 2021. Scalable and interpretable semantic change detection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4642–4652, Online. Association for Computational Linguistics.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Vlad Niculae and Andre Martins. 2020. Lp-sparsemap: Differentiable relaxed optimization for sparse structured prediction. In *International Conference on Machine Learning*, pages 7348–7359. PMLR.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Nina Poerner, Hinrich Schütze, and Benjamin Roth. 2018. Evaluating neural network explanation methods using hybrid documents and morphosyntactic agreement. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 340–350, Melbourne, Australia. Association for Computational Linguistics.

Marco Ribeiro, Sameer Singh, and Carlos Guestrin. 2016a. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 97–101, San Diego, California. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016b. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Shruti Rijhwani and Daniel Preotiuc-Pietro. 2020. Temporally-informed analysis of named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7605–7617, Online. Association for Computational Linguistics.

Guy D Rosin and Kira Radinsky. 2022. Temporal attention for language models. *arXiv preprint arXiv:2202.02093*.

Paul Röttger and Janet Pierrehumbert. 2021. Temporal adaptation of BERT and performance on downstream document classification: Insights from social media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2400–2412, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jeffrey C Schlimmer and Richard H Granger. 1986. Incremental learning from noisy data. *Machine learning*, 1(3):317–354.

Sofia Serrano and Noah A Smith. 2019a. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951.

Sofia Serrano and Noah A. Smith. 2019b. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153. PMLR.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. 2021. We need to talk about random splits. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832, Online. Association for Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR.

Marcos Treviso and André F. T. Martins. 2020. The explanation game: Towards prediction explainability through sparse communication. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 107–118, Online. Association for Computational Linguistics.

Adam Tsakalidis, Marya Bazzi, Mihai Cucuringu, Pierpaolo Basile, and Barbara McGillivray. 2019. Mining the UK web archive for semantic change detection. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1212–1221, Varna, Bulgaria. INCOMA Ltd.

Uriel Weinreich, William Labov, and Marvin Herzog. 1968. *Empirical foundations for a theory of language change*. University of Texas Press.

Gerhard Widmer and Miroslav Kubat. 1993. Effective learning in dynamic environments by explicit context tracking. In *European Conference on Machine Learning*, pages 227–243. Springer.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 260–267, Rochester, New York. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28.

A Data split statistics

For X-FACT, FactCheck, AmazDigiMu and AmazPantry, we take the earliest 80% data as the training and testing set for Syn, the earlier 80% to 90% data as Asy1 and the rest (i.e. the latest 10%) as Asy2. For Yelp and AGNews, which have more available data for each year, we sample a same number of data from each year based on the year with the least data available. This allows year-wise analysis. We take the newer 2 years of data as the Asy datasets, per year per Asy. We also experiment with the original dataset (OSyn, including Syn, Asy1 and Asy2) to provide a comparison. Table 2 shows details of each split for each task.

B Models and Hyper-parameters

Feature attributions We use BERT-base with pre-trained weights from the Huggingface library (Wolf et al., 2020). We use the AdamW optimizer (Loshchilov and Hutter, 2017) with an initial learning rate of $1e^{-5}$ for fine-tuning BERT and $1e^{-4}$ for the fully-connected classification layer. We train our models for 3 epochs using a linear scheduler, with 10% of the data in the first epoch as warming up. We also use a grad-norm of 1.0. The model with the lowest loss on the development set is selected. All models are trained across 5 random seeds, and we report the average and standard deviation.

FRESH For the rationale extractor, we use the same model for extracting rationales from feature attributions. For the classifier (trained only on the extracted rationales), we also use BERT-base with the same optimizer configuration and scheduler warm-up steps. We use a grad-norm of 1.0 and select the model with the lowest loss on the development set. We train across 5 random seeds for 5 epochs.

HardKUMA We use the 300-dimensional pre-trained GloVe embeddings from the 840B release (Pennington et al., 2014) and keep them frozen. Similar to Bastings et al. (2019) and Chrysostomou and Aletras (2022a), we use a Bi-LSTM layer of 200-d for the rationale extractor. We use the Adam optimizer (Loshchilov and Hutter, 2017) with a learning rate between $1e^{-3}$ and $1e^{-5}$ and a weight decay of $1e^{-5}$. We also enforce a grad-norm of 5.0 and train for 20 epochs across 5 random seeds. Following Guerreiro and Martins (2021), we select the model with the highest F1-macro score on the development set and tuning the Lagrangian relaxation algorithm parameters between $1e^{-2}$ and $1e^{-5}$.

SPECTRA Following Guerreiro and Martins (2021), we take the 300-dimensional pre-trained GloVe embeddings from the 840B release (Pennington et al., 2014) as word representations and keep them frozen. As Guerreiro and Martins (2021) suggested, results with Bi-LSTM layers were competitive with those with BERT reported in Jain et al. (2020). We, therefore, instantiate all encoder networks as Bi-LSTM layers of hidden size 200. We also use the AdamW optimizer (Loshchilov and Hutter, 2017) for training SPECTRA. We use a learning rate $\in [1e^{-3}, 5e^{-4}, 1e^{-4}, 5e^{-5}]$ and l_2 regularization $\in [1e^{-4}, 1e^{-5}]$ for the training. We also use a grad-norm of 5.0. We train all models for highlights extraction for a minimum of 3 epochs and maximum of 20 epochs. For matching extraction, we set the number of minimum epochs of 3 and maximum epochs of 10. We implement early stopping for the model training if F1 stop increasing over 5 epochs and for highlights extraction if F1 stop increasing over 3 epochs. Tributes are paid to Guerreiro and Martins (2021) for the published code published.

All experiments are run on a single NVIDIA Tesla V100 GPU.

C Post-hoc Explanations Faithfulness

D Select-then-predict Predictive Performance

D.1 HardKUMA and SPECTRA Predictive Performance

D.2 FRESH Predictive Performance

Task	Label Num	Domain	Start Date	End Date	Time Span(Days)	Median Date	Interquartile Time Span (Days)	Data Num
AGNews	4	OSyn Full	2004-08-18	2008-02-20	1281	2006-09-27	980	46790
		OSyn Train	2004-08-18	2008-02-20	1281	2006-09-27	979	28074
		OSyn Test	2004-08-18	2008-02-20	1281	2006-09-28	983	9358
		Syn Train	2004-08-18	2006-12-20	854	2005-01-20	648	9358
		Syn Test	2004-08-18	2006-12-20	854	2005-01-20	650	9358
		Asy1 Test	2007-01-30	2007-12-31	335	2007-06-29	191	9358
		Asy2 Test	2008-01-01	2008-02-20	50	2008-01-24	25	9358
X-FACT	6	OSyn Full	2007-01-04	2018-11-12	4330	2013-06-27	1727	12050
		OSyn Train	2007-01-04	2018-11-12	4330	2013-08-19	1769	9639
		OSyn Test	2007-04-26	2018-11-07	4213	2013-09-19	1727	1206
		Syn Train	1995-04-01	2016-08-31	7823	2012-08-30	1296	7232
		Syn Test	2007-01-04	2016-08-31	3527	2012-08-17	1320	1204
		Asy1 Test	2016-08-31	2017-09-30	395	2017-02-24	216	1205
		Asy2 Test	2017-09-30	2018-11-12	408	2018-05-10	209	1204
FactCheck	2	OSyn Full	1995-09-25	2021-07-19	9429	2016-12-29	1519	12640
		OSyn Train	1996-02-27	2021-07-19	9274	2016-12-29	1527	10086
		OSyn Test	1995-09-25	2021-06-23	9403	2016-12-20	1515	1277
		Syn Train	1995-09-25	2019-05-01	8619	2016-04-14	1437	7446
		Syn Test	1996-08-02	2019-05-01	8307	2016-03-09	1319	1241
		Asy1 Test	2019-05-02	2020-05-15	379	2019-11-18	182	1368
		Asy2 Test	2020-05-15	2021-07-19	430	2020-10-12	132	1368
AmazDigiMu	3	OSyn Full	1998-07-09	2018-09-26	7384	2015-01-19	789	169623
		OSyn Train	1998-07-09	2018-09-26	7384	2015-01-20	788	135698
		OSyn Test	1998-08-21	2018-09-20	7335	2015-01-08	794	16962
		Syn Train	1998-08-21	2016-05-07	6469	2014-09-20	673	101774
		Syn Test	1998-12-20	2016-05-07	6351	2014-09-18	669	16963
		Asy1 Test	2016-05-07	2016-12-30	237	2016-08-12	120	16962
		Asy2 Test	2016-12-30	2018-09-26	635	2017-08-07	290	16962
AmazPantry	3	OSyn Full	2006-04-09	2018-10-04	4561	2016-09-27	485	137611
		OSyn Train	2006-04-09	2018-09-28	4555	2016-09-27	486	110088
		OSyn Test	2006-10-14	2018-10-04	4373	2016-09-24	474	13761
		Syn Train	2006-04-28	2017-07-30	4111	2016-07-06	413	82566
		Syn Test	2006-12-22	2017-07-30	3873	2016-07-08	406	13762
		Asy1 Test	2017-07-30	2018-01-21	175	2017-10-16	92	13761
		Asy2 Test	2018-01-21	2018-10-04	256	2018-04-12	94	13761
Yelp	5	OSyn Full	2005-02-16	2022-01-19	6181	2014-01-02	3274	15372
		OSyn Train	2005-02-16	2022-01-19	6181	2014-01-30	3277	11956
		OSyn Test	2005-02-16	2022-01-19	6181	2013-09-18	3297	1708
		Syn Train	2005-02-16	2018-12-31	5066	2012-01-16	2556	8540
		Syn Test	2005-02-16	2018-12-24	5059	2011-10-22	2457	1708
		Asy1 Test	2019-01-01	2020-12-31	730	2020-01-01	375	1708
		Asy2 Test	2021-01-01	2022-01-19	383	2022-01-01	195	1708

Table 2: Summary of the original dataset and the chronological splits for each task.

Task	Train Set	Test Set	Fulltext F1	Normalised Sufficiency								Normalised Comprehensiveness							
				$\alpha\nabla\alpha$	α	DL	$x\nabla x$	lime	IG	DLsp	Gsp	$\alpha\nabla\alpha$	α	DL	$x\nabla x$	lime	IG	DLsp	Gsp
AGNews	Original	Original	90.8	1.53	1.34	1.00	1.31	0.99	1.32	1.04	1.00	2.42	2.09	1.23	1.87	1.20	1.88	0.98	0.96
	Syn	Syn	89.1	1.51	1.36	0.93	1.32	0.98	1.31	0.98	0.97	2.95	2.18	1.36	2.00	1.22	1.97	1.01	1.03
		Asy1	87.5	1.48	1.37	0.97	1.29	1.06	1.29	1.02	0.97	2.96	2.35	1.36	2.00	1.43	1.97	1.03	1.01
		Asy2	85.5	1.49	1.40	0.97	1.26	1.02	1.25	1.02	0.97	3.08	2.51	1.41	2.04	1.38	1.99	1.02	1.02
X-FACT	Original	Original	37.4	1.24	1.12	0.88	1.14	1.04	1.15	0.95	0.98	1.67	1.40	1.12	1.35	1.06	1.37	1.00	0.98
	Syn	Syn	38.2	1.16	1.02	0.91	1.02	1.03	1.03	0.93	0.95	1.64	1.02	1.04	1.22	1.05	1.19	0.90	0.96
		Asy1	37.4	1.15	0.99	0.98	1.03	1.03	1.04	0.95	0.95	1.74	1.03	1.08	1.26	1.06	1.24	1.04	1.00
		Asy2	38.2	1.14	0.98	0.96	1.02	1.03	1.02	0.94	0.95	1.58	0.92	1.00	1.15	1.01	1.10	0.93	0.92
FactCheck	Original	Original	74.4	2.00	1.73	1.10	1.09	1.19	1.11	1.03	0.98	4.67	3.42	1.26	2.23	1.12	2.27	1.10	1.07
	Syn	Syn	71.1	2.8	2.4	1.17	1.41	1.06	1.48	1.13	1.03	2.85	2.67	1.52	1.95	0.96	1.96	1.16	1.08
		Asy1	70.6	1.15	1.2	0.94	0.96	1.04	1.00	0.91	0.88	0.99	1.12	0.84	1.10	1.03	0.94	1.15	1.01
		Asy2	73.5	1.26	1.13	1.00	1.22	1.24	1.23	1.08	0.96	1.06	1.18	0.83	1.00	1.00	0.91	1.04	0.98
AmazDigiMu	Original	Original	71.4	3.19	2.97	1.81	1.89	0.58	1.96	1.45	1.07	1.61	1.45	1.16	1.21	0.63	1.19	1.03	1.13
	Syn	Syn	69.3	1.31	1.25	0.51	0.83	0.55	0.71	0.95	0.77	2.71	1.93	1.55	1.76	1.08	2.11	0.92	1.19
		Asy1	63.7	1.19	1.15	0.48	0.84	0.57	0.74	0.93	0.81	2.63	1.74	1.23	1.82	1.09	2.28	0.80	1.19
		Asy2	58.2	1.16	1.14	0.49	0.84	0.58	0.74	0.94	0.82	2.52	1.63	1.10	1.73	1.05	2.14	0.75	1.18
AmazPantry	Original	Original	71.0	2.40	1.03	0.97	1.15	0.88	1.37	0.82	0.78	1.99	1.08	0.94	1.60	1.19	1.48	0.91	1.25
	Syn	Syn	69.6	2.18	1.55	0.82	1.47	0.57	1.71	0.82	0.97	2.14	1.68	1.03	1.62	0.76	1.55	0.77	0.85
		Asy1	68.2	2.14	1.54	0.81	1.51	0.59	1.73	0.79	1.00	2.13	1.63	1.00	1.62	0.77	1.55	0.75	0.87
		Asy2	68.1	2.06	1.46	0.76	1.52	0.56	1.74	0.76	0.98	2.12	1.58	0.96	1.64	0.73	1.57	0.72	0.86
Yelp	Original	Original	61.4	2.58	1.73	1.16	1.62	0.97	1.60	1.03	0.99	2.27	1.56	1.13	1.64	0.76	1.61	1.05	1.01
	Syn	Syn	60.3	2.16	1.61	1.30	1.38	0.95	1.37	1.12	0.96	2.52	1.76	1.34	1.65	0.85	1.66	1.10	0.99
		Asy1	59.2	2.39	1.70	1.39	1.58	0.85	1.59	1.15	0.96	2.30	1.62	1.39	1.67	0.83	1.69	1.11	0.99
		Asy2	61.0	2.39	1.69	1.36	1.60	0.88	1.61	1.13	0.95	2.27	1.60	1.31	1.63	0.80	1.65	1.07	0.95

Table 3: AOPC Normalized Sufficiency and Comprehensiveness (higher is better) for the OSyn, Syn and Asy of 8 feature attribution approaches on their TOPK tokens. Each feature is presented as the ratio to the random attribution baseline.

Task	Domain	LSTM F1	LSTM std	KUMA F1	KUMA std	KUMA len	SPECTRA F1	SPECTRA std
AGNews	OSyn	86.8	0.1	85.0	0.4	23.2	83.2	0.4
	Syn	84.4	0.1	81.9	1.1	39.1	80.4	0.8
	Asy1	82.4	0.5	79.1	1.7	36.8	78.3	1.7
	Asy2	79.4	0.6	77.4	1.5	37.0	77.7	1.9
X-FACT	OSyn	35.4	1.5	11.7	1.8	41.1	24.9	1.1
	Syn	34.3	0.5	9.9	0.0	43.6	22.0	5.7
	Asy1	31.6	3.0	12.7	0.0	42.5	16.7	7.8
	Asy2	30.0	1.9	12.5	0.0	41.6	20.6	7.3
FactCheck	OSyn	62.4	2.9	47.2	1.7	53.9	50.7	5.3
	Syn	55.6	3.2	47.3	2.7	78.5	50.5	6.5
	Asy1	51.7	2.8	45.2	2.9	78.9	45.5	4.2
	Asy2	49.8	3.6	41.9	2.3	79.0	43.2	6.4
AmazDigiMu	OSyn	66.8	0.8	64.9	1.7	19.0	48.7	1.4
	Syn	66.4	1.0	65.8	1.7	18.2	43.0	2.9
	Asy1	59.1	0.9	56.9	1.3	18.7	37.6	3.6
	Asy2	54.4	1.2	52.1	1.2	18.6	39.0	3.6
AmazPantry	OSyn	67.5	0.4	64.4	0.4	18.4	58.1	0.5
	Syn	67.4	0.6	63.5	1.0	17.9	48.9	1.4
	Asy1	66.4	0.4	62.7	0.9	18.7	50.0	1.1
	Asy2	67.0	0.9	62.7	1.1	19.6	51.5	1.0
Yelp	OSyn	48.0	0.4	28.8	1.6	11.7	36.2	2.2
	Syn	39.1	1.3	19.4	0.5	11.3	32.8	6.5
	Asy1	43.3	1.4	20.1	0.4	14.9	33.6	4.4
	Asy2	43.2	1.8	20.0	0.8	15.0	34.7	5.8

Table 4: Averaged macro F1 performance and standard deviation over five runs for HardKUMA and SPECTRA and their corresponding full-text models (T-test is conducted between HardKUMA and LSTM, and between SPECTRA and LSTM, for each split.

Task	Domain	BERT F1	BERT std	$\alpha\nabla\alpha$ F1	$\alpha\nabla\alpha$ std	DL F1	DL std	$x\nabla x$ F1	$x\nabla x$ std
AGNews	OSyn	90.8	0.3	90.1	0.1	89.7	0.1	83.1	0.3
	Syn	89.1	0.5	87.0	0.6	86.7	0.4	81.5	0.6
	Asy1	87.5	0.7	84.6	0.8	83.6	0.9	75.0	1.3
	Asy2	85.5	0.5	83.3	0.3	81.6	0.9	73.7	1.7
X-FACT	OSyn	37.4	2.9	29.9	4.2	35.8	2.2	29.0	1.3
	Syn	38.2	2.4	32.5	1.7	28.3	5.1	27.2	2.0
	Asy1	37.4	1.8	32.6	2.8	21.4	2.4	18.3	2.2
	Asy2	38.2	1.7	30.0	2.6	21.6	2.9	24.4	1.4
FactCheck	OSyn	74.4	3.2	77.1	0.4	83.9	0.3	64.2	3.0
	Syn	71.1	1.9	71.8	0.2	64.7	1.0	53.2	4.3
	Asy1	70.6	2.7	72.1	0.0	67.0	1.3	45.9	2.4
	Asy2	73.5	1.7	74.2	0.1	65.7	1.5	42.8	2.7
AmazDigiMu	OSyn	71.4	1.6	66.2	0.8	68.1	0.5	50.8	2.2
	Syn	69.3	2.5	63.6	0.9	70.9	0.5	51.2	4.3
	Asy1	63.7	1.5	52.5	0.6	61.7	1.2	49.3	3.1
	Asy2	58.2	1.1	50.2	0.9	56.3	1.0	44.6	1.7
AmazPantry	OSyn	71.0	0.5	66.2	0.5	68.6	0.2	49.0	1.9
	Syn	69.6	1.5	62.2	1.0	54.2	1.4	49.2	2.7
	Asy1	68.2	1.4	58.9	0.9	52.3	0.9	47.5	2.6
	Asy2	68.1	2.5	57.8	0.9	51.0	0.7	47.4	2.8
Yelp	OSyn	61.4	1.3	58.7	0.8	51.5	1.7	45.3	1.4
	Syn	60.3	0.6	56.5	0.4	52.2	1.5	43.4	4.3
	Asy1	59.2	1.3	55.0	0.8	52.4	0.8	46.1	2.4
	Asy2	61.0	0.7	58.5	0.7	54.0	1.4	45.5	3.7

Table 5: Averaged macro F1 performance and standard deviation over five runs for HardKUMA and SPECTRA and their corresponding full-text models.