Comparing Automatic and Human Evaluation of Local Explanations for Text Classification

Dong Nguyen♠◇
♠The Alan Turing Institute, London
◇School of Informatics, University of Edinburgh, Edinburgh
dnguyen@turing.ac.uk

Abstract

Text classification models are becoming increasingly complex and opaque, however for many applications it is essential that the models are interpretable. Recently, a variety of approaches have been proposed for generating local explanations. While robust evaluations are needed to drive further progress, so far it is unclear which evaluation approaches are suitable. This paper is a first step towards more robust evaluations of local explanations. We evaluate a variety of local explanation approaches using automatic measures based on word deletion. Furthermore, we show that an evaluation using a crowdsourcing experiment correlates moderately with these automatic measures and that a variety of other factors also impact the human judgements.

1 Introduction

While the impact of machine learning is increasing rapidly in society, machine learning systems have also become increasingly complex and opaque. Classification models are usually evaluated based on prediction performance alone (e.g., by measuring the accuracy, recall, and precision) and the interpretability of these models has generally been undervalued. However, the importance of interpretable models is increasingly being recognized (Doshi-Velez and Kim, 2017; Freitas, 2014).

First, higher interpretability could lead to more effective models by revealing incompleteness in the problem formalization (Doshi-Velez and Kim, 2017), by revealing confounding factors that could lead to biased models, and by supporting error analyses or feature discovery (Aubakirova and Bansal, 2016). Second, with the increasing adoption of machine learning approaches for humanities and social science research, there is also an increasing need for systems that support exploratory analyses and theory development.

Various approaches have been explored to increase the interpretability of machine learning models (Lipton, 2016). This paper focuses on *local* explanation, which aims to explain the prediction for an individual instance (e.g., Ribeiro et al. (2016)). A study by Herlocker et al. (2000) found that providing local explanations could help improve the acceptance of movie recommendation systems. Local explanations can come in different forms. For example, Koh and Liang (2017) identify the most influential training documents for a particular prediction. The most common type of local explanation involves identifying the important parts of the input for a prediction, such as the most predictive words in a document for a text classification model.

In this paper we focus on local explanations for text classification. Below is a fragment of a movie review. The words identified by a local explanation method to explain a neural network prediction are in bold. The review is labeled with a negative sentiment, but the classifier incorrectly predicted a positive sentiment. The highlighted words help us understand why.

> steve martin is one of the **funniest** men alive. if you can take that as a **true** statement, then your disappointment at this film will equal mine. martin can be **hilarious**, **creating** some of the best laugh-out-loud **experiences** that have ever taken place in movie theaters. you won't find any of them here. [...]

Words such as *funniest* and *hilarious* were important for the prediction. Besides providing evidence *for* a predicted label, some local explanations can also provide evidence *against* a predicted label. For example, in the above example, the word *disappointment* was one of the highest ranked words against the predicted label.

Proceedings of NAACL-HLT 2018, pages 1069–1078
New Orleans, Louisiana, June 1 - 6, 2018. ©2018 Association for Computational Linguistics

Ineffective approaches could generate misleading explanations (Lipton, 2016), but evaluating local explanations is challenging. A variety of approaches has been used, including only visual inspection (Ding et al., 2017; Li et al., 2016a), intrinsic evaluation approaches such as measuring the impact of deleting the identified words on the classifier output (Arras et al., 2016), and user studies (Kulesza et al., 2015).

Contributions To further progress in this area, it is imperative to have a better understanding of how to evaluate local explanations. This paper makes the following contributions:

- *Comparison of local explanation methods for text classification.* We present an in-depth comparison between three local explanation approaches (and a random baseline) using two different automatic evaluation measures on two text classification tasks (Section 4).
- *Automatic versus human evaluation.* Automatic evaluations, such as those based on word deletions, are frequently used since they enable rapid iterations and are easy to reproduce. However, it is unclear to what extent they correspond with human-based evaluations. We show that the automatic measures correlate moderately with human judgements in a task setting and that other factors also impact human judgement. (Section 5).

2 Related Work

Research on interpretable machine learning models has so far mainly focused on computer vision systems (e.g., Simonyan et al. (2013)). Topic modeling is one of the exceptions within NLP where the interpretability of models has been important, since topic models are often valued for their interpretability and are integrated in various user interfaces (Paul, 2016). There has recently been an increasing interest in improving the interpretability of NLP models, perhaps driven by the increasing complexity of NLP models and the rise of deep learning (Manning, 2015).

Global approaches aim to provide a global view of the model. One line of work involves making the machine learning model itself more interpretable, e.g., by enforcing sparsity or imposing monotonicity constraints (Freitas, 2014). However, often there is a trade-off between accuracy and interpretability as adding constraints to the model could reduce the performance. An alternative involves extracting a more interpretable model, such as a decision tree, from a model that is less interpretable, such as a neural network (Craven, 1996). In this case, model performance is not sacrificed but it is essential that the proxy is faithful to the underlying model.

However, often a machine learning model is so complex that interpretable, trustworthy global explanations are difficult to attain. *Local* explanations aim to explain the output for an individual instance. For some models the local explanations are relatively easy to construct, e.g., displaying the word probabilities of a Naive Bayes model with respect to each label (Kulesza et al., 2015) or displaying the path of a decision tree (Lim et al., 2009). However, these models may not be easily interpretable if they make use of many features.

For many machine learning models, extracting local explanations is even less straight-forward. Proposed approaches so far include using the gradients to visualize neural networks (Aubakirova and Bansal, 2016; Li et al., 2016a; Simonyan et al., 2013), measuring the effect of removing individual words (or features) (Li et al., 2016b; Martens and Provost, 2014), decomposition approaches (Arras et al., 2016; Ding et al., 2017), and training an interpretable classifier (e.g., linear model) that approximates the neighborhood around a particular instance (Ribeiro et al., 2016).

Some approaches have only been evaluated using visual inspection (Ding et al., 2017; Li et al., 2016a). Goyal et al. (2016) identified important words for a visual question answering system and informally evaluated their approach by analyzing the distribution among PoS tags (e.g., assuming that nouns are important). However, quantitative evaluations are needed for more robust comparisons. Such evaluations have included measuring the impact of the deletion of words identified by the explanation approaches on the classification output (Arras et al., 2016, 2017), or testing whether the explanation was consistent with an underlying gold model (Ribeiro et al., 2016). These automatic evaluations are fast to carry out but act as a simplistic proxy for explanation quality. While a few user studies have been performed to evaluate explanations (e.g., Ribeiro et al. (2016)), we are not aware of work that analyzes how automatic evaluation measures compare to human-based evaluation.

3 Experimental Setup

This section describes the datasets, the classification models and the local explanation approaches used in our experiments.

3.1 Datasets

We experiment with two datasets (Table 1):

- **Twenty newsgroups (20news).** The Twenty Newsgroups dataset has been used in several studies on ML interpretability (Arras et al., 2016; Kapoor et al., 2010; Ribeiro et al., 2016). Similar to Ribeiro et al. (2016), we only distinguish between *Christianity* and *Atheism*. We use the **`20news-bydate`** version, and randomly reserve 20% of the training data for development.

- **Movie reviews.** Movie reviews with polarity labels (positive versus negative sentiment) from Pang and Lee (2004). We use the version from Zaidan et al. (2007). The dataset is randomly split into a train (60%), development (20%) and test (20%) set.

	Movie	**20news**
# training docs	1072	870
# development docs	358	209
# test docs	370	717
label distribution (pos. class)	50.00%	44.49%

Table 1: Dataset statistics

3.2 Text Classification Models

We experiment with two different models. Logistic Regression (**`LR`**) is implemented using Scikit-learn (Pedregosa et al., 2011) with Ridge regularisation, unigrams and a TF-IDF representation, resulting in a 0.797 accuracy on the movie dataset and a 0.921 accuracy on the 20news dataset. We experiment with a LR model, because the contributions of individual features in a LR model are known. We thus have a ground truth for feature importance to compare against for this model. We also use a feedforward neural network (**`MLP`**) implemented using Keras (Chollet et al., 2015), with 512 hidden units, ReLU activation, dropout (0.5, not optimized) and Adam optimization, resulting in a 0.832 accuracy on the movie dataset and a 0.939 accuracy on the 20news dataset.

3.3 Local Explanation Methods

In this paper, we focus on local explanation approaches that identify the most influential parts of the input for a particular prediction. In this paper we limit our focus to individual words for explaining the output of text classification models. Other representations, e.g., explanations using phrases or higher-level concepts are left for future work. We experiment with explanations for the *predicted* class, since in real-life settings usually no ground truth labels are available. We experiment with the following local explanation approaches:

- **Random.** A random selection of words in the document.

- **LIME** (Ribeiro et al., 2016) is a model-agnostic approach and involves training an interpretable model (in this paper, a linear model with Ridge regularisation) on samples created around the specific data point by perturbing the data. We experiment with 500–5000 samples and use the implementation provided by the authors.[1]

- **Word omission.** This approach aims to estimate the contribution of individual words by deleting them and measuring the effect, e.g., by the difference in probability (Robnik-Šikonja and Kononenko, 2008). Within NLP, variations have been proposed by Kádár et al. (2016), Li et al. (2016b) and Martens and Provost (2014). It is also similar to occlusion in the context of image classification, which involves occluding regions of the input image (Zeiler and Fergus, 2014). For **`LR`**, this approach corresponds to ranking words according to the regression weights (and considering the frequency in the text) and is therefore optimal. For **`MLP`**, we use the difference in probability for the predicted class ($\hat{y}$) when removing word w from input $\boldsymbol{x}$: $p(\hat{y}|\boldsymbol{x}) - p(\hat{y}|\boldsymbol{x}_{\setminus w})$. This approach supports explanations based on interpretable features (e.g., words) even when the underlying representation may be less interpretable. However note that in general, this omission approach might not be optimal, since it estimates the contribution of words independently. This approach is also computationally expensive, especially when many features are used.

[1] `https://github.com/marcotcr/lime`.

- **First derivative saliency.** This approach computes the gradient of the output with respect to the input (e.g., used in Aubakirova and Bansal (2016), Li et al. (2016a) and Simonyan et al. (2013)). The obtained estimates are often referred to as saliency values. Several variations exist, e.g., Li et al. (2016a) take the absolute value. In this paper, the raw value is taken to identify the words important for and against a certain prediction.

4 Automatic Evaluation

In this section we explore automatic evaluation of local explanations. Local explanations should exhibit high *local fidelity*, i.e. they should match the underlying model in the neighborhood of the instance (Ribeiro et al., 2016). An explanation with low local fidelity could be misleading. Because we generate explanations for the predicted class (rather than the ground truth), explanations with high local fidelity do not necessarily need to match human intuition, for example when the classifier is weak (Samek et al., 2017). Ideally, the evaluation metrics are model agnostic and do not require information that may not always be available such as probability outputs. This paper focuses on local fidelity, but other aspects might also be desired, such as sparsity (Samek et al., 2017; Ribeiro et al., 2016; Martens and Provost, 2014).

4.1 Evaluation Metrics

We measure local fidelity by deleting words in the order of their estimated importance for the prediction. Arras et al. (2016) generated explanations with the correct class as target. By deleting the identified words, accuracy increased for incorrect predictions and decreased for correct predictions. However, their approach assumes knowledge of the ground-truth labels.

We take an alternative, but similar, approach. Words are also deleted according to their estimated importance, e.g. $w_1 ... w_n$ with w_1 the word with the highest importance score, but for the *predicted class* instead. For each document, we measure the number of words that need to be deleted before the prediction switches to another class (the ***switching point***), normalized by the number of words in the document. For example, a value of 0.10 indicates that 10% of the words needed to be deleted before the prediction changed. An advantage of this approach is that ground-truth labels are not needed and that it can be applied to black-box classifiers, we only need to know the predicted class. Furthermore, the approach acts on the raw input. It requires no knowledge of the underlying feature representation (e.g., the actual features might be on the character level). We also experiment with the measure proposed by Samek et al. (2017), referred to as the area over the perturbation curve (***AOPC***):

$$AOPC = \frac{1}{K+1} \langle \sum_{k=1}^{K} f(\boldsymbol{x}) - f(\boldsymbol{x}_{\backslash 1..k}) \rangle_{p(x)}$$

where $f(\boldsymbol{x}_{\backslash 1..k})$ is the probability for the predicted class when words $1..k$ are removed and $\langle \cdot \rangle_{p(x)}$ denotes the average over the documents. This approach is also based on deleting words, but it is more fine-grained since it uses probability values rather than predicted labels. It also enables evaluating negative evidence. A drawback is that AOPC requires access to probability estimates of a classifier. In this paper, K is set to 10.

For **LR**, the exact contribution of individual features to a prediction is known and the words in the document that contributed most to the prediction can be computed directly. For this classifier, the optimal approach corresponds to the omission approach.

4.2 Results

Table 3 reports the results by measuring the effect of word deletions and reporting the average switching point. Lower values indicate that the method was better capable of identifying the words that contributed most towards the predicted class, because on average fewer words needed to be deleted to change a prediction. Table 2 shows the AOPC values with a cut-off at 10. We measure AOPC in two settings: removing positive evidence (higher values indicate a more effective explanation) and negative evidence (lower values indicate a more effective explanation).

Comparison local explanation methods As expected, LIME improves consistently when more samples are used. Furthermore, when comparing the scores of the omission approach for the **LR** model (which corresponds to the ground-truth) we observe that LIME with 5000 samples comes close to the optimal score. We use the two-tailed paired permutation test to test for significance between all methods with both evaluation measures. In al-

	20news (topic)				Movie (sentiment)			
	LR		MLP		LR		MLP	
	pos.	neg.	pos.	neg.	pos.	neg.	pos.	neg.
random	0.0116	0.0101	0.0110	0.0112	0.0073	0.0112	0.0083	0.0066
LIME-500	0.1855	-0.0301	0.1279	-0.0266	0.3168	-0.0786	0.2125	-0.0727
LIME-1000	0.2013	-0.0303	0.1350	-0.0268	0.3509	-0.0793	0.2330	-0.0738
LIME-1500	0.2067	-0.0302	0.1369	-0.0269	0.3586	-0.0794	0.2375	-0.0740
LIME-2000	0.2092	-0.0304	0.1378	-0.0269	0.3628	-0.0794	0.2394	-0.0740
LIME-5000	0.2128	-0.0303	0.1391	-0.0270	0.3693	-0.0794	0.2425	**-0.0741**
omission	**0.2342**	**-0.0307**	**0.1422**	-0.0272	**0.3724**	**-0.0795**	**0.2440**	**-0.0741**
saliency	-	-	0.1418	**-0.0273**	-	-	0.2439	**-0.0741**

Table 2: AOPC results. For each method, AOPC is used to evaluate the words identified to be supportive of the predicted class (positive evidence) and words identified to be supportive of the other class (negative evidence). For LIME, results are reported for different sample sizes.

	20news		Movie	
	LR	MLP	LR	MLP
random	0.8617	0.8880	0.6586	0.6843
LIME-500	0.4394	0.5330	0.1747	0.1973
LIME-1000	0.3098	0.4164	0.0811	0.1034
LIME-1500	0.2607	0.3566	0.0613	0.0800
LIME-2000	0.2336	0.3235	0.0547	0.0743
LIME-5000	0.1895	0.2589	0.0474	0.0664
omission	**0.1595**	0.2662	**0.0449**	0.0644
saliency	-	**0.2228**	-	**0.0639**

Table 3: The % of words that needs to be deleted to change the prediction (the switching point).

method	SP	AOPC
random	0.581	-0.168
LIME-500	0.932	-0.897
LIME-1000	0.884	-0.877
LIME-1500	0.863	-0.872
LIME-2000	0.850	-0.870
LIME-5000	0.826	-0.866
omission	0.814	-0.865
saliency	0.812	-0.865

Table 4: Spearman correlation between prediction confidence and AOPC and the switching point (SP) for the **`MLP`** classifier on the movie dataset.

most all cases, the differences are highly significant ($p < 0.001$), except the difference in average switching point between the omission and salience approach on the movies dataset with the **`MLP`** classifier (n.s.) and the difference in average switching point between the omission and LIME-5000 approach on 20news with the **`MLP`** classifier (n.s.). The difference in AOPC scores for evaluating negative evidence was not significant in many cases.

Metric sensitivity First, the results suggest that the values obtained depend strongly on the type of task and classifier. The explanation approaches score better on the sentiment detection task in both Tables 2 and 3. For example, fewer words need to be removed on average to change a prediction in the movie dataset (Table 3). A possible explanation is that for sentiment detection, a few words can provide strong cues for the sentiment (e.g., *terrific*), while for (fine-grained) topic detection (e.g., distinguishing between *Christianity* and *atheism*) the evidence tends to be distributed among more words. Better values are also obtained for the **`LR`** classifier (a linear model) than for **`MLP`**.

Second, as shown in Table 2, AOPC enables assessing negative evidence (i.e. the words that provide evidence for the opposite class). The obtained absolute values are much smaller compared to the values obtained for the words identified as positive evidence. This is expected, since the positive evidence in a document for the predicted class should be larger than the negative evidence.

Third, we analyze the relation between the word deletion evaluation measures and the prediction confidence of the classifiers, based on the probability of the output class. Table 4 reports the Spearman correlations for the **`MLP`** classifier on the movie dataset (similar trends were observed with the **`LR`** classifier). There is a strong correlation between the prediction confidence and the word deletion evaluation measures. The higher the prediction confidence of a classifier, the more words need to be deleted before a prediction changes (e.g., see the switching points). However, the strength of the correlations is lower for the more robust explanation methods (LIME-5000, omission and saliency).

5 Human-based Evaluation

In the previous section we evaluated the local explanation approaches using automatic measures. However, the explanations are meant to be presented to *humans*. We therefore turn to evaluating the explanations using crowdsourcing. We analyze the usefulness of the generated explanations in a task setting and analyze to what extent the automatic measures correspond to the human-based evaluations. The crowdsourcing experiments are run on CrowdFlower. Only crowdworkers from Australia, Canada, Ireland, United Kingdom and the United States and with quality levels two or three were accepted.

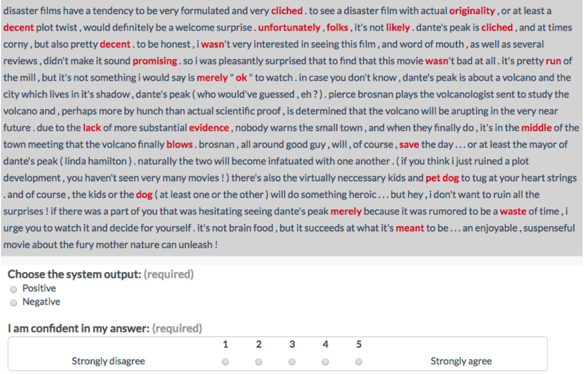
5.1 Forward Prediction Task

One way to evaluate an explanation is by asking humans to guess the output of a model based on the explanation and the input. Doshi-Velez and Kim (2017) refer to this as forward simulation/prediction. As mentioned by Doshi-Velez and Kim (2017), this is a simplified task. Evaluations using more specific application-oriented tasks or tailored towards specific user groups should be explored in future work. We have chosen the forward prediction task as a first step since it is a general setup that could be used to evaluate explanations for a variety of tasks and models.

In this study, crowdworkers are shown the texts (e.g., a movie review), in which the top words identified by the local explanation approaches are highlighted. Crowdworkers are then asked to guess the output of the system (e.g., a positive or negative sentiment). The crowdworkers are also asked to state their confidence on a five-point Likert scale (‘*I am confident in my answer*’: strongly disagree . . . strongly agree).

Note that the workers need to guess the output of the model regardless of the true label (i.e. the model may be wrong). The crowdworkers are therefore presented with documents with different prediction outcomes (true positive, true negative, false negative, and false positive). We sample up to 50 documents for each prediction outcome. A screenshot is shown in Figure 1. A quiz and test questions are used to ensure the quality of the crowdworkers. Instructions as well as the test questions included cases where the system made an incorrect prediction, so that workers understood that the task was different than standard labeling tasks. See Appendix A for more details.

We experiment with the following parameters: *methods* (random baseline, LIME with 500 and 5000 samples, word omission, saliency) and the *number of words* (10, 20). We experiment with both datasets. Due to space constraints, we only experiment with the **MLP** classifier. We collected the data in August and September 2017. Each HIT (Human Intelligence Task) was carried out by five crowdworkers. We paid $0.03 per judgement. On the 20news dataset, we collected 7,200 judgements from 406 workers (mean nr of. judgements per worker: 17.73, std.: 7.21) and on the movie dataset we collected 8,100 judgements from 445 workers (mean nr of. judgements per worker 18.20, std: 7.24).



disaster films have a tendency to be very formulated and very **cliched** . to see a disaster film with actual **originality** , or at least a **decent** plot twist , would definitely be a welcome surprise . **unfortunately** , **folks** , it's not **likely** . dante's peak is **cliched** , and at times corny , but also pretty **decent** . to be honest , i **wasn**'t very interested in seeing this film , and word of mouth , as well as several reviews , didn't make it sound **promising** . so i was pleasantly surprised that to find that this movie **wasn**'t bad at all . it's pretty **run** of the mill , but it's not something i would say is **merely** " **ok** " to watch . in case you don't know , dante's peak is about a volcano and the city which lives in it's shadow , dante's peak (who would've guessed , eh ?) . pierce brosnan plays the volcanologist sent to study the volcano and , perhaps more by hunch than actual scientific proof , is determined that the volcano will be arupting in the very near future . due to the **lack** of more substantial **evidence** , nobody warns the small town , and when they finally do , it's in the **middle** of the town meeting that the volcano finally **blows** . brosnan , all around good guy , will , of course , **save** the day . . . or at least the mayor of dante's peak (linda hamilton) . naturally the two will become infatuated with one another . (if you think i just ruined a plot development , you haven't seen very many movies !) there's also the virtually neccessary kids and **pet dog** to tug at your heart strings . and of course , the kids or the **dog** (at least one or the other) will do something heroic . . . but hey , i don't want to ruin all the surprises ! if there was a part of you that was hesitating seeing dante's peak **merely** because it was rumored to be a **waste** of time , i urge you to watch it and decide for yourself . it's not brain food , but it succeeds at what it's **meant** to be . . . an enjoyable , suspenseful movie about the fury mother nature can unleash !

Choose the system output: (required)

○ Positive

○ Negative

I am confident in my answer: (required)

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Figure 1: Screenshot of the task

Confidence Most workers chose confidence values of three or four. Table 6 reports the confidence scores by method. On the movie dataset, the trends match the intrinsic evaluations closely. The random method leads to the lowest confidence score, followed by LIME-500 and LIME-5000, and explanations from the omission and saliency approach both lead to the highest confidence scores. On the 20news dataset, the trends are less clear. We observe a small, significant negative correlation between confidence values and time spent (Spearman correlation: ρ=-0.08, p <0.0001 on the movie dataset, ρ=-0.06, p <0.0001 on 20news).

Accuracy Table 6 also reports the fraction of correct guesses per method. Random explanations lead to the lowest accuracies, followed by LIME with 500 samples. The differences between LIME-5000, omission and saliency are small and not consistent across datasets. The crowd had a higher accuracy on the movie data, except when the explanations were randomly generated.

			TP			TN			FP			FN	
Method	**#w**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**
						Movies							
random	10	0.652	3.42	250	0.484	3.35	250	0.581	3.26	155	0.355	3.53	155
LIME-500	10	0.848	3.65	250	0.796	3.58	250	0.787	3.41	155	0.710	3.61	155
LIME-5000	10	0.900	3.73	250	0.896	3.70	250	0.852	3.43	155	0.748	3.63	155
omission	10	0.932	3.80	250	0.916	3.67	250	0.845	3.52	155	0.781	3.54	155
saliency	10	0.940	3.87	250	0.872	3.78	250	0.819	3.50	155	0.729	3.59	155
random	20	0.628	3.48	250	0.512	3.43	250	0.471	3.24	155	0.374	3.45	155
LIME-500	20	0.864	3.65	250	0.784	3.51	250	0.742	3.54	155	0.794	3.39	155
LIME-5000	20	0.880	3.76	250	0.864	3.63	250	0.787	3.77	155	0.800	3.67	155
omission	20	0.896	3.95	250	0.884	3.72	250	0.832	3.54	155	0.761	3.58	155
saliency	20	0.860	3.70	250	0.876	3.78	250	0.819	3.63	155	0.806	3.57	155
						20news							
random	10	0.664	3.45	250	0.656	3.45	250	0.489	3.44	45	0.514	3.47	175
LIME-500	10	0.724	3.53	250	0.768	3.73	250	0.733	3.62	45	0.817	3.84	175
LIME-5000	10	0.740	3.52	250	0.832	3.87	250	0.556	3.29	45	0.697	3.75	175
omission	10	0.652	3.37	250	0.800	3.78	250	0.689	3.31	45	0.754	3.63	175
saliency	10	0.712	3.42	250	0.832	3.77	250	0.689	3.80	45	0.789	3.86	175
random	20	0.616	3.52	250	0.696	3.57	250	0.511	3.84	45	0.537	3.65	175
LIME-500	20	0.668	3.50	250	0.788	3.67	250	0.689	3.22	45	0.697	3.73	175
LIME-5000	20	0.720	3.52	250	0.888	3.86	250	0.667	3.36	45	0.709	3.60	175
omission	20	0.692	3.53	250	0.864	3.80	250	0.644	3.42	45	0.726	3.71	175
saliency	20	0.752	3.64	250	0.904	3.74	250	0.711	3.67	45	0.783	3.78	175

Table 5: Results forward prediction task, with the accuracy (acc), average confidence (conf) and the number of judgements (n). The results are separated according to TP (true positive), TN (true negative), FP (false positive) and FN (false negative) predictions, and the number of words shown (#w).

method	**accuracy**		**confidence**	
	20news	**movie**	**20news**	**movie**
random	0.616	0.522	3.520	3.402
LIME-500	0.740	0.798	3.640	3.555
LIME-5000	0.761	0.851	3.665	3.673
omission	0.744	0.868	3.615	3.694
saliency	0.790	0.851	3.691	3.701

Table 6: Confidence and accuracy results

Table 5 separates the results by the different prediction outcomes. The results suggest that false positive and false negative are the most revealing. In these cases, crowdworkers are not able to rely on their intuition and a strong explanation should convince them that the system makes a mistake. Otherwise, crowd workers might choose the label matching the document (and not necessarily the classifier output). This is especially salient in the 20news dataset, where the random approach performs better than expected on the true positives and true negatives. For example, compare the random approach with the omission approach on true positives with ten word explanations.

Our experiments also show that local explanations in the form of the most predictive words are sometimes not enough to simulate the output of a system. For example, the best accuracy on true positive instances in the 20news data is only 0.752. The movie dataset contains difficult instances as well. For example, the omission method identifies the following words in a movie review to explain a false positive prediction: '*believes*', '*become*', '*hair*', '*unhappy*', '*quentin*', '*directed*', '*runs*', '*filled*', '*fiction*', '*clint*'. Due to the composition of the training data, the system has associated words like '*quentin*' and '*clint*' with a positive sentiment. This may have confused the crowdworkers as most of them guessed incorrectly. Expanding the explanation with for example influential documents (Koh and Liang, 2017) or a visualization of the class distributions of the most influential words could make the explanations more informative.

Correlation with automatic evaluation For each explanation, we compute the fraction of workers who correctly predicted the classifier output (the 'crowd accuracy') and correlate these with the automatic measures. We expect a negative correlation with the switching points and a positive correlation with the AOPC. The correlations are moderate (Table 8). The correlations with AOPC on the movie data are the biggest on the false positives and false negatives, when workers are not able to rely on their intuition. The correlations

		TP			TN			FP			FN		
Noise	**AOPC**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**	**Acc**	**Conf**	**n**
0	0.2627	0.940	3.87	250	0.872	3.78	250	0.819	3.50	155	0.729	3.59	155
0.2	0.2044	0.896	3.60	250	0.780	3.67	250	0.735	3.39	155	0.735	3.58	155
0.4	0.1485	0.824	3.62	250	0.776	3.68	250	0.723	3.37	155	0.645	3.31	155
0.6	0.0851	0.800	3.40	250	0.756	3.40	250	0.710	3.63	155	0.639	3.34	155
0.8	0.0411	0.736	3.29	250	0.640	3.35	250	0.632	3.25	155	0.523	3.25	155

Table 7: Forward prediction task with noisy explanations on the movie dataset and the saliency method

	Movie		20news	
	SP	**AOPC**	**SP**	**AOPC**
tp	−0.144**	0.156***	0.134**	−0.161***
fn	−0.283***	0.367***	−0.181***	0.343***
tn	−0.195***	0.153***	−0.203***	−0.027
fp	−0.076	0.290***	−0.076	0.172

Table 8: Spearman correlation between automatic measures and crowd accuracy. Significance: $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

Dependent variable: crowd accuracy	
Switching point	-0.365*** (0.023)
Classifier confidence	0.344*** (0.044)
Prediction outcome: fp	0.053** (0.021)
Prediction outcome: tn	0.093*** (0.020)
Prediction outcome: tp	0.132*** (0.019)
Constant	0.472*** (0.037)
R^2: 0.177 (Adj.: 0.174)	
F Stat.: 69.255*** (df = 5; 1614)	

Table 9: OLS results with switching points on the movie data (n = 1,620). *p<0.1; **p<0.05; ***p<0.01. Prediction outcome base level = fn.

measured on the true positives in 20news are opposite of what we expect. The 20news data is noisy and the classifier picks up on spurious features, possibly confusing the workers.

An example in the 20news data is an e-mail with the following words highlighted: '*thank*', '*mail*', '*discussions*', '*seminary*', '*before*', '*thanks*', '*question*', '*fill*', '*affected*', '*during*', '*proofs*'. The classifier was confident and the computed switchpoint was low. The e-mail comes from the atheism newsgroup, which becomes clear from reading the text. The highlighted words are all more likely to occur in the christianity newsgroup, but on their own they are not intuitive to lay people. Consequently, workers guessed incorrectly that the predicted label was atheism. Explanations that also show the negative evidence (in this case, words such as '*atheism*' and '*atheists*') and/or the word distributions across classes would likely have led to better crowd accuracy.

Dependent variable: crowd accuracy	
AOPC	0.543*** (0.042)
Classifier confidence	0.395*** (0.048)
Prediction outcome: fp	0.079*** (0.021)
Prediction outcome: tn	0.119*** (0.020)
Prediction outcome: tp	0.171*** (0.020)
Constant	0.222*** (0.046)
R^2: 0.133 (Adj.: 0.130)	
F Stat.: 49.572*** (df = 5; 1614)	

Table 10: OLS results with AOPC on the movie data (n = 1,620). *p<0.1; **p<0.05; ***p<0.01. Prediction outcome base level = fn.

As shown in section 4, the automatic measures correlate strongly with the prediction confidence of the classifier. More words need to be removed before a prediction changes (i.e. a higher switching point) when the classifier is more confident. However, we also find that higher classifier confidence leads to higher crowd accuracies (e.g., ρ = 0.236, $p < 0.001$ on the 20news dataset). We therefore fit an Ordinary Least Squares (OLS) model to control for these different factors (Table 9), with crowd accuracy as the dependent variable. A higher switching point significantly leads to a lower accuracy. However, classifier confidence and prediction outcome also significantly impact the accuracy. Similar trends are observed for the AOPC measure (Table 10). We also find that the automatic evaluation measures significantly impact crowd accuracy on the 20news dataset, but the patterns are less strong.

Noise In our final experiment we analyze the effect of noise. We focus on explanations based on saliency scores on the movie dataset. We experiment with introducing noise to the top ten words (Table 7) and we collect additional judgements. A noise level of 0.2 indicates that two out of the top ten words are randomly replaced by other words. The results show that with increasing the noise, as expected, both the performance and average AOPC score decrease.

6 Conclusion

There has been an increasing interest in improving the interpretability of machine learning systems, but evaluating the quality of explanations has been challenging. This paper focused on evaluating local explanations for text classification. Local explanations were generated by identifying important words in a document for a prediction. We compared automatic evaluation approaches, based on measuring the effect of word deletions, with human-based evaluations. Explanations generated using word omissions and first derivatives both performed well. LIME (Ribeiro et al., 2016) performed close to these methods when using enough samples. Our analyses furthermore showed that the evaluation numbers depend on the task/dataset and the confidence of the classifiers.

Next, crowd workers were asked to predict the output of the classifiers based on the generated explanations. We found moderate, but significant, correlations between the automatic measures and crowd accuracy. In addition, the human judgements were impacted by the confidence of the classifier and the type of prediction outcome (e.g., a false negative versus a true positive). Our results also suggest that only highlighting words is sometimes not enough. An explanation can highlight the most important parts of an input and score well on automatic measures, but if the explanation is not intuitive (for example due to biases in the data), humans are still not able to predict the output.

For the classification tasks in this paper (topic classification and sentiment detection) individual words are often predictive. As a result, local explanation approaches that select words independently worked well. However, we expect that for tasks where individual words are not predictive, the current evaluation methods and local explanation approaches may not be sufficient. Furthermore, in future work more fine-grained visualizations (e.g., Handler et al. (2016)) could be explored.

Acknowledgements

This work was supported by The Alan Turing Institute under the EPSRC grant EP/N510129/1. The author is supported with an Alan Turing Institute Fellowship (TU/A/000006). This work was supported with seed funding award SF023.

References

Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Muller, and Wojciech Samek. 2016. Explaining predictions of non-linear classifiers in NLP. In *Proceedings of the 1st Workshop on Representation Learning for NLP*. pages 1–7.

Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. 2017. ”What is relevant in a text document?”: An interpretable machine learning approach. *PLOS ONE* 12(8):e0181142.

Malika Aubakirova and Mohit Bansal. 2016. Interpreting neural networks to improve politeness comprehension. In *Proceedings of EMNLP 2016*. pages 2035–2041.

Chollet et al. 2015. Keras. https://github.com/fchollet/keras.

Mark W. Craven. 1996. *Extracting comprehensible models from trained neural networks*. Ph.D. thesis, University of Wisconsin–Madison.

Yanzhuo Ding, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Visualizing and understanding neural machine translation. In *Proceedings of ACL 2017*. pages 1150–1159.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. In *arXiv preprint arXiv:1702.08608*.

Alex A. Freitas. 2014. Comprehensible classification models: A position paper. *SIGKDD Explorations Newsletter* 15(1):1–10.

Yash Goyal, Akrit Mohapatra, Devi Parikh, and Dhruv Batra. 2016. Towards transparent AI systems: Interpreting visual question answering models. In *International Conference on Machine Learning (ICML) Workshop on Visualization for Deep Learning*.

Abram Handler, Su Lin Blodgett, and Brendan O'Connor. 2016. Visualizing textual models with in-text and word-as-pixel highlighting. In *Proceedings of the 2016 Workshop on Human Interpretability in Machine Learning*.

Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of CSCW '00*. pages 241–250.

Ákos Kádár, Grzegorz Chrupala, and Afra Alishahi. 2016. Representation of linguistic form and function in recurrent neural networks. *CoRR* abs/1602.08952.

Ashish Kapoor, Bongshin Lee, Desney Tan, and Eric Horvitz. 2010. Interactive optimization for steering machine classification. In *Proceedings of CHI '10*. pages 1343–1352.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of ICML 2017*. pages 1885–1894.

Todd Kulesza, Margaret Burnett, Weng-Keen Wong, and Simone Stumpf. 2015. Principles of explanatory debugging to personalize interactive machine learning. In *Proceedings of IUI '15*. pages 126–137.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016a. Visualizing and understanding neural models in NLP. In *Proceedings of NAACL 2016*. pages 681–691.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016b. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220* .

Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. 2009. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of CHI '09*. pages 2119–2128.

Zachary C. Lipton. 2016. The mythos of model interpretability. In *Proceedings of the 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016)*. pages 96–100.

Christopher D. Manning. 2015. Computational linguistics and deep learning. *Computational Linguistics* 41(4):701–707.

David Martens and Foster Provost. 2014. Explaining data-driven document classifications. *MIS Quarterly* 38(1):73–100.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity. In *Proceedings of ACL*. pages 271–278.

Paul. 2016. Interpretable machine learning: Lessons from topic modeling. In *Proceedings of the CHI Workshop on Human-Centered Machine Learning*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of KDD '16*. pages 1135–1144.

Marko Robnik-Šikonja and Igor Kononenko. 2008. Explaining classifications for individual instances. *IEEE Transactions on Knowledge and Data Engineering* 20(5):589–600.

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

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR* abs/1312.6034.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Proceedings of NAACL 2007*. pages 260–267.

Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *ECCV 2014*. pages 818–833.

A Appendix: Crowdsourcing

Test questions were manually selected and were cases for which there should be no doubt about the correct answer (e.g., a simple movie review with only words such as '*brilliant*', '*terrific*', etc. highlighted). Thus, these are questions where workers would only fail if they did not pay attention or if they did not understand the task. Explanations were provided for most test questions and were shown after an answer was submitted. The test questions contained instances with different prediction outcomes (e.g. false positives and true positives) to make the task clear. To make sure that the test questions did not overlap with the actual HITs (which were generated to explain the predictions of the **MLP**), the test questions were explanations generated for the **LR** classifier.

A quiz with test questions was provided to the crowdworkers when starting the task. If the workers performed poorly on the quiz, they were not allowed to continue with the task. Throughout the task, test questions were entered in between the actual HITs (one out of five presented HITs was a test question), to monitor the quality and to flag crowdworkers who performed poorly. We closely monitored the responses to the test questions and in the pilot phase we did remove a few that turned out not be suitable. In the final task, workers performed overall very well on the test questions.

The task was consistently rated positive by the crowdworkers. The task was divided into several batches and the overall rating was consistently above 4.5 (out of 5). The payment rating was consistently above 4. The tasks explicitly mentioned that the results will be used for scientific research ('*By participating you agree that these results will be used for scientific research.*').