Evaluation on attribution-based explanation of TextCNN

Xiong, W.

Award date:
2018

Technische Universiteit
Eindhoven
University of Technology

Department of Mathematics and Computer Science
Data Mining Group

Evaluation on Attribution-based Explanation of TextCNN

Master Thesis

Wenting Xiong

Supervisors:
prof.dr. M. Pechenizkiy (TU/e)
W. van Ipenburg MSc (Rabobank)

Assessment Committee:
prof.dr. M. Pechenizkiy (TU/e)
W. van Ipenburg MSc (Rabobank)
dr. ir. George Fletcher (TU/e)

Submit version

Eindhoven, August 2018

Abstract

In this thesis, attribution-based explanation approaches on text classification problem with TextCNN is investigated. Two explanation approaches were investigated: Layer-wise Relevance Propagation (LRP) and saliency map. The research problem is formulated to develop an evaluation framework with clearly defined standards (i.e., measurable), and to develop a visualization application which can help analysts to locate the reasons for the model's predictions.

The evaluations on attribution-based explanations were carried out in three different aspects: word level, embedded-document level and n-gram level explanations.

On word level, document representations were generated with vector representations for words, weighted with attribution scores on words. Qualitatively, a PCA was performed on the document representations. The results were plotted in 2-dimensional space with the documents' class as labels. Compared to the results on unweighted document representations, clear clusters were formed when attribution scores were applied. Quantitatively, several different classifiers were employed to the document representations. Compared to the unweighted document representations, the model performance on weighted document representations was vastly improved.

On embedded-document level, each embedding column was treated as a feature. By removing the features with the largest positive or negative impact, the model accuracy decreased drastically when using LRP attribution scores. By removing the least important features, the model accuracy was not compromised with both LRP and saliency map more than randomly removing features. The differences of attributions generated for different classes were also analyzed. The experiments were able to demonstrate that by removing features based on the differences, we were able to "guide" the misclassification towards one particular class.

On n-gram level, each convolutional filter represents an n-gram feature. Thus, removing a filter from the model is equivalent to removing the contributions from an n-gram feature. By removing filters with the largest positive or negative impact, the model accuracy dropped significantly when using LRP attribution scores. By removing the least important filters, the model accuracy did not decrease more than randomly removing filters. From the experiments on attribution differences for different classes, the same observation was made that the misclassification was guided towards one class.

In the visualization application, explanations on both word feature and n-gram features were applied. The differences in attributions for each feature were also visualized to offer a more detailed explanation. A misclassification example was analyzed to show how attribution scores are useful in explaining the model's prediction.

Preface

This thesis is the result of the master graduation project for Data Science Engineering at Eindhoven University of Technology. The research of this project is performed within the Data Mining Group of the TU/e in collaboration with Rabobank located in the Netherlands.

First of all, I would like to take this opportunity to thank my supervisor Prof. Mykola Pechenizkiy, who had offered me valuable advice when I was feeling lost. Also, I would like to thank the PHDs, Tita and Simon who followed closely on my project and gave me abundant practical suggestions. Tita has inspired me with some ideas and helped me realize to what extent this project should proceed. Simon has given me useful suggestions for evaluating the explanation methods. The experiences at Rabobank is a valuable journey as well. I am grateful for both Werner van Ipenburg and Jan W. Veldsink for their help in our weekly meetings. Every time we have a discussion, they would offer me suggestions on how to improve my experiments and thesis.

I would like to thank all my friends and family for their unconditional support during my master's project.

Contents

Chapter 1

Introduction

In this chapter, we first introduce the motivation for the project for the necessity of model explainability. Then, we briefly summarize the goals and challenges in evaluating an explanation method. In accordance with the challenges, we describe our approaches and results as well. In the last section, we give the outline of this thesis.

1.1 Motivation

Neural networks have attained near-human accuracy performances in various scenarios such as image classification [8], text classification [6] and speech recognition [5]. Applying neural networks in industries is the inevitable future. However, the networks continue to be treated as black boxes. A level of trust in the models is required if these networks are to be incorporated into critical processes such as banking, medical, planning and control [11]. The lack of explainability is one of the largest barriers to making use of neural networks in a banking system. For banks, it is not sufficient to have a high-performance machine learning model. More importantly, a convincing explanation of the results should also be offered so that humans can have a better understanding of the model's decisions.

As one of the largest banks in the Netherlands, Rabobank is dealing with a massive amount of text data from the clients or employees. Based on advanced deep learning algorithms, it is possible to perform classification, summarization, segmentation and various other tasks to gain insights into the data. As discussed above, a certain trust between human and deep learning algorithms must be established. To achieve such a goal, an extensive evaluation of explanation techniques is conducted to show their potential future use for the company. We chose text classification as the focused problem set as it is a fundamental topic for natural language processing.

1.2 Goals and Challenges

The techniques used to generate explanations in this project are attribution-based, which means a score is assigned to each input feature to assess their influence on the predictions. The overall goal for this thesis is to develop an evaluation framework for the attribution-based explanations to reveal how the explanation approaches work, to validate the correctness of the explanations and to demonstrate the usefulness of the explanations in real life.

The overall challenge in the evaluation of explanations is to offer a more in-depth understanding of

the model's decisions as well as how the decisions are explained. If the explanation technique itself is black-box to human, instead of offering information on the neural networks, they can only result in more confusion. When using an explanation technique on complicated models, it is crucial that the explaining technique achieves transparency to a certain degree.

To dive into details, the challenges can be elaborated regarding model, data and the classification task:

1. A neural network model usually consists of many layers. The outcome of each layer can be considered as the input for the next. Thus, an explanation can be generated for each of the layers. However, an explanation on intermediate layers showing how important each neuron is to the predictions can hardly make sense to the human. Therefore, approaches are needed to reflect the explanations of intermediate layers onto the original input, which makes sense to the human.

2. To evaluate the influences of a data point on the outcome, it is useful to remove some features to show how much the outcome changes. In this project, only text data is used. Removing a word from the text could result in a different prediction. However, it could also be the contribution of other factors. For example, the changes in the semantic meanings or the sentence length could have contributed to the prediction changes. Besides, for documents of different length, removing the same number of words results in removing a different proportion of inputs. Thus, the results of word-deleting experiments are not very trustworthy.

3. When explaining the results of a text classification task, the challenges can be summarized into two parts:

   - For a general classification task, it is essential to understand what the attributions mean. Also, the differences in approaches should be demonstrated through experiments.
   - For a multi-class classification task, explanations per class should be offered. When an input is classified as "A", instead of explaining why it belongs to "A", explanations on why it is not classified as "B" should also be offered.

1.3 Approaches and Results

1. Approaches

   A summarization of our approaches to evaluate the attribution scores is provided here corresponding to the challenges.

   - To demonstrate the explanations on the intermediate layers, we have visualized the attributions onto the text input as n-gram features. A TextCNN is used as the target model to explain. Its intermediate layers' outputs are higher (n-gram) representations of the input data.
   - To avoid removing words, we adopt a different strategy. Based on the attribution scores on each input feature (word) and the vector representation of each word, a document representation can be generated. A word considered more important has a larger weight in producing the document representation. Then, we perform an evaluation of the document representations to investigate whether words relevant to the output are identified.
   - To assess the influence of feature on the outputs without removing words, we have performed the feature removing experiments on the intermediate layers including the Embedding layer and Convolutional layer.

2. Results

   Our experimental results on two publicly available datasets suggest that when using weights on each word to generate document representations, the approach using weights can generate representations more relevant to the classification result than the approach without weights. For the feature removing experiments, the results demonstrate that the differences in attribution scores towards different class reflect the reason why the model is in favor of a particular class over another. In addition, the differences in explanation approaches are also evident through the experiments.

   For presenting the potential of explanations in business, an interactive visualization application was built to help human to understand the reasons for the model's outputs. The visualization includes explanations of both words and n-gram features. Moreover, it offers local explanations of why the model is in favor of one class over another. The visualization is built based on a multi-class classification problem.

1.4 Thesis Outline

The thesis is organized as follows:

- In Chapter 2 (Background & Related work), we first overview the word embedding and TextCNN, which are the prerequisite for the explanation approaches. Then we discuss the main state-of-art attribution-based explanation approaches as well as the current evaluation approach, which are the focus of the thesis.
- In Chapter 3 (Problem Statement), firstly, we introduce the business problem and analyze the limitations of the current applications of attribution-based explanations. The goals and challenges to solve this problem are also described. Secondly, we give a formal description of the attribution-based explanation. We have identified the weakness in their evaluation and proposed several research questions to be answered.
- In Chapter 4 (Explanation Evaluation Framework), we will establish an understanding of what is considered a good evaluation for attribution-based approaches at first. Then, we propose three different evaluation approaches aiming at various aspects to evaluate the attribution-based explanations, as well as a detailed description of how to proceed with the corresponding experiments.
- In Chapter 5 (Interactive Visualization Demo), we develop an application as a demo to present how the attribution scores can be used in business.
- In Chapter 6 (Experiments), first of all, we perform an exploratory study on the dataset. Secondly, we describe the settings and results on the TextCNN models. Next, we conduct experiments including embedded feature removing, filter removing and assessment on document representation on all dataset. We present and discuss the results to support our arguments. Last but not least, an analysis using the visualization tool is conducted as an example.
- In Chapter 7 (Conclusions), we conclude both academic and business contributions of the thesis. Furthermore, we also discuss the limitations and future work.

Chapter 2

Background & Related work

In this chapter, we first introduced word embedding and TextCNN, which are the prerequisite for our study. Then, we discussed the strength of attribution-based explanation approaches over perturbation-based approaches on neural network models. Last but not least, the insufficiency in the current evaluation method on attribution-based explanations are discussed

2.1 Word Embedding

Word embedding is a feature learning technique that can represent words with a list of real numbers. The idea was first mentioned to resolve the curse of dimensionality by learning a distributed representation for words which allows each training sentence to inform the model about an exponential number of semantically neighboring sentences [4]. Such representation can capture semantic similarities, relationships between words through relative positions of the corresponding vectors.

There are various types of word embeddings. In this thesis, Global Vectors for Word Representation (GloVe) will be applied [13]. It is used to capture the meaning of one word embedding with the structure of the whole observed corpus; word frequency and co-occurrence counts are the primary measures on which the majority of unsupervised algorithms are based on. GloVe model trains on global co-occurrence counts of words and makes sufficient use of statistics by minimizing least-squares error and, as a result, producing a word vector space with meaningful substructure. Such an outline sufficiently preserves words similarities with vector distance.

2.2 Neural Network in Text Classification

Text classification is a fundamental topic for natural language processing, in which one needs to assign predefined categories to free-text documents. The range of text classification research goes from designing the best features to choosing the best possible machine learning classifiers. Convolutional Neural Network(CNN) is a typical architecture of neural network. CNN is useful in extracting position invariant features, thus widely used in classifying images. When applied on text data, TextCNN was developed to obtain the n-gram features.

TextCNN [7]: In recent years, CNN is frequently explored in NLP tasks. A series of experiments with CNN were conducted on pre-trained word vectors for text classification tasks. It was shown that a simple CNN with little hyper-parameter tuning achieves excellent results on multiple

benchmarks. Additionally, a modification on the architecture to allow the use of task-specific and static vector was made and proved to have improved on several sentiment analysis and question classification tasks. The proposed model contains two channels of word vectorsone that is kept static throughout training and one that is fine-tuned via backpropagation. The architecture is illustrated in Figure 2.2.

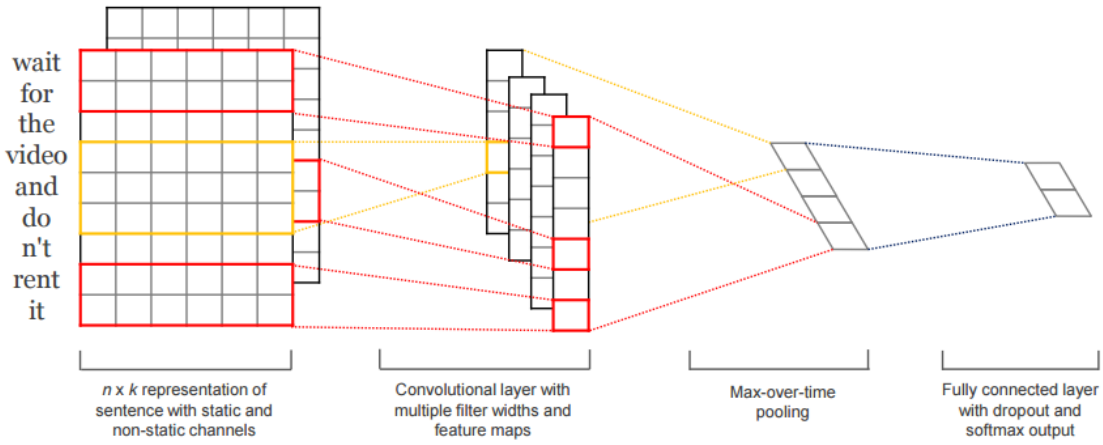


Figure 2.1: Model architecture for CNN for an example sentence [7]

2.3 Model Interpretability

Interpretability is not a monolithic concept. The techniques and model properties proposed to improve model interpretability broadly fall into two categories: ante-hoc transparency and post-hoc explanations [10]. Ante-hoc approaches choose a model that is already interpretable and train it. While post-hoc approaches select a model that already works well and develop a technique to interpret its predictions. In this thesis, only post-hoc explanations are involved.

Approaches that focus on post-hoc interpretability extract information from learned models. Even though post-hoc interpretations do not always explain precisely how a model works, they still present useful information. In this section, attribution-based approaches will be discussed. These approaches assign an attribution value, which can also be seen as an “importance“ or “relevance“ value, to each input feature of a network.

There are two different types of attribution-based approaches that are often applied to real life problem: perturbation-based approaches and attribution-based approaches.

2.3.1 Perturbation-based Approach

Perturbation-based methods compute the attribution of an input feature (or set of features) by removing, masking or altering them, and running a forward pass on the new input, measuring the difference with the original output. This technique has been applied to CNNs in the domain of image classification [17], visualizing the probability of the correct class as a function of the position of a grey patch occluding part of the image.

However, such methods can be computationally inefficient as each perturbation requires a separate forward propagation through the network. They may also underestimate the importance of features that have saturated their contribution to the output[15].

2.3.2 Attribution-based Approach

The approaches introduced here are backpropagation-based methods that compute the attributions for all input features in a single forward and backward pass through the network.

Saliency map:

Saliency map is proposed to constructs attributions by taking the absolute value of the partial derivative of the target output with respect to the input features[16]. The explanations generated with saliency map are global, which means that for a particular feature (pixel/word), the attribution in all inputs is the same. The drawback of this approach is that it fails to determine whether a feature has a positive or negative impact on the outcome.

Layer-wise relevance propagation (LRP):

LRP is a technique for determining which features in a particular input vector contribute most strongly to a neural networks output. In a document classification task [2], LRP measures how much each word in a text contributes to the decision of a CNN classifier and an SVM classifier by backward-propagating its output. 'Model explanatory power' was defined to evaluate and compare the outcomes based on an extrinsic validation procedure. The results show that while two models perform similarly in terms of classification accuracy, the decisions of the CNN are based on fewer, more semantically meaningful words. Besides CNN, LRP was also modified to apply to recurrent neural network[9]. LRP was modified to be used on more types of neural network layers, including many-to-one weighted linear connections, and two-to-one multiplicative interactions.

In contrast to perturbation methods, backpropagation approaches are computationally efficient as they propagate an importance signal from the output neuron backwards through the layers towards the input in a single pass [1].

In this thesis, backpropagation-based approaches will be investigated due to their computational efficiency.

2.3.3 Evaluation of Attribution-based Approach

To evaluate the quality of an attribution-based approach in identifying relevant features, a word-deleting method was established and applied on attribution scores generated by saliency map and LRP on CNNs [3]. By deleting the words with the highest attribution scores, a more drastic drop in model accuracy was observed when using the attribution scores generated by LRP and saliency map compared to randomly.

The word-deleting method is very intuitive and straightforward. However, more factors are contributing to the accuracy changes. Removing a word not only eliminates the contribution of this particular word but also affects the contributions of other words since the model's decisions are usually made based on the combination of words.

Chapter 3

Problem Statement

3.1 Business Problem

The current application of attribution-based explanations is to calculate the attribution of each word for the correct output. By highlighting the positive or negative attributions of each word, analysts/decision makers can inspect the reasons for the model's decision.

However, based on the architecture of TextCNN, the model's decisions are made based on the combination of a few words. The drawback of only highlighting individual words is that it fails to capture the mutual contribution of the words. For example, for a prase "not a very good movie", the sentiment classification result should be that it is a negative statement. But it is likely that only the attribution score for the word "not" is negative while "a very good movie" has a positive attribution score. This example may appear straightforward. But in more complicated contexts, explanations on a higher level (closer to the output in the model) should offer more insights on how the model works.

In addition, in a classification task, the decisions are made by comparing the output probability for all classes. Attribution scores based on a wrong class may also be interesting to an analyst as it reveals some information on how classification can go wrong.

To resolve this problem, explanations of both words and n-gram features should be generated and visualized for all outputs.

Objective:

- For a multi-class classification task, develop a visualization tool to explain the model output
- Generate explanations towards all classes on both words and n-gram features. We will discuss the approach to generate explanations on n-gram feature in Section 4.4.
- The tool should be able to explain why a document is classified as "A" instead of B, in addition to explaining why it is classified as "A". We will discuss such explanations in Section 4.3 and Section 4.4.

3.2 Research Problem

We describe the research problem in two parts: generate attribution scores and evaluate attribution scores.

3.2.1 Generate Attribution Scores

For generating attribution scores, a unified framework of gradient-based attribution methods for neural networks is used here [1].

Problem Formulation

In this section, a formal explanation will be given about assigning an attribution value to an input feature. It is necessary to point out that an input feature can be from any layer of the network. For example, in a text classification problem, a word can be considered an input feature. An element from the vector representation of a word can also be a feature. Even the input for a neuron in an intermediate layer can be an input feature.

For an input $x = [x_1, x_2, ..., x_N]$, the model produces an output $f(x) = [f_1(x), ...f_C(x)]$, where C is the number of output neurons (number of classes). For a particular class c, the task is to produce attribution scores $R^c = [R_1^c, ..., R_N^c]$ for each input feature x_i.

For saliency map, which takes the absolute value of the partial derivative, the attribution scores are computed according to the below formula[16]:

$$R_i^c = \left\| \frac{\partial}{\partial x_i} f^c(x) \right\|$$

LRP attribution scores are computed with one backward pass on the network. The method redistributes the prediction score $f_c(x)$ layer by layer until reaching the desired layer[14]:

$$R_j^c = \sum_k \frac{x_j w_{jk}}{\sum_j x_j w_{jk}} R_k^c$$

The following rule holds for attribution scores from all layers that for a particular class c, the sum of attribution scores on a layer is equal to the prediction score $f_c(x)$:

$$\sum_i R_i^c = ... = \sum_j R_j^c = \sum_k R_k^c = ... = f_c(x)$$

The fundamental differences (characteristics) in saliency map and LRP is summarized as below:

Method	Signed	Use Input in Backpropagation
Saliency map	No	No
LRP	Yes	Yes

Table 3.1: Characteristics of Saliency Map and LRP

3.2.2 Evaluation on Attribution Scores

A conventional approach to evaluate the explanations on text data is to delete the most/least important words from each document and observe the impact on model accuracy. When removing the most important words, the model accuracy is expected to drop drastically. When deleting the

[1] https://github.com/marcoancona/DeepExplain

least important words, the model performance should not be compromised more than randomly deleting a word. Some experimental results are consistent with the above expectation[14].

However, other factors also contribute to this outcome that are hard to be eliminated. For example, after removing words, the document length is changed. The new document might not be semantically meaningful anymore. Moreover, for documents of different length, the impact of deleting one word is very different. In addition, some of the n-gram information is also lost. The above reasons can all result in accuracy drop. Therefore, more robust evaluation approaches are needed to improve the trustworthiness of the explanations.

Therefore, the research questions will focus on the evaluation of the attribution scores:

1. Can we evaluate the attributions without deleting words?

   When processing text data, pre-trained embeddings are usually seen as the features. While deleting words, which is equivalent to removing rows in the word embedding, is not the ideal evaluation method, it is also possible to consider eliminating features on the column axis.

2. Can we evaluate the attributions without making any changes to the input data?

   The attribution-based approaches discussed here are all backpropagation-based. Therefore, attribution scores can be generated on every layer of the neural network. When preserving all the features, changes can be made to nodes in the neural network. In this way, attribution scores for higher representations of features can be evaluated as well.

3. Can we evaluate the attributions without making any changes to the data or the model?

   As attribution scores quantify the features' contribution to the outcome, the features with high attribution scores should be able to represent the classification results better. Therefore, we can utilize the attribution scores as weights on each feature to generate representations for the document. If the attribution scores are highly relevant to the classifications, the information in a document towards its class would be amplified, resulting in a better clustering/classification results on the document representations.

4. How good is the evaluation approach?

   To answer this question, a clear definition of what consists of a good evaluation of explanation approaches. Through experimental results, we will assess what the evaluation methods succeeded or failed to achieve.

Based on our research questions, a framework will be established to evaluate the attribution scores on multiple levels.

Chapter 4

Explanation Evaluation Framework

In this chapter, we will first define metrics for a good evaluation framework. Then, we will make the assumptions for our experiments. Next, a framework that thoroughly evaluates the attribution scores will be introduced. The evaluation will be conducted on the word level, embedded-document level and n-gram level.

4.1 Assumptions and Definitions

a) *Assumptions.*

In the following sections, the evaluation approaches involve removing features to determine their influence on the model outputs. Theoretically speaking, when a feature is removed, the LRP attribution scores on other features will not stay the same, while the saliency map attribution scores remain unchanged. The reason is that saliency map does not compute the attribution scores based on the input (feature). To make sure that every time a feature with maximum/minimum attribution score is removed, the attribution score should be computed right before removing this particular feature. To avoid the massive amount of computations, the following assumption is made.

Assumption 1.: when a feature is removed, though the LRP attribution scores on other features are different, the ranking of the rest of the features will still stay the same.

b) *Definition for a good evaluation framework.*

A good evaluation for attribution-based approaches should be able to:

- Evaluate on measurable objectives: clear, measurable metrics should be adopted for comparing the evaluation results.
- Evaluate whether the truly important features can be found: truly important features have high relevance/contribution to the output. By amplifying or diminishing a feature's influence on the model, we should be able to observe accuracy changes.
- Collect and analyze information on characteristics of the explanation approaches: based on the results, the characteristics about explanation approaches should be made evident. For example, LRP is signed while saliency map is not, the results should be able to reflect it.

4.2 Evaluation on Word-level

To evaluate the attribution scores on word level, an approach that is fundamentally different than removing features is adopted. A document representation can be generated by taking the mean of the representations of the words. Here, the attribution score on each word is used as weights to generate the document representation. For document x_i which consists of words with representation $[v_1, ..., v_j, .., v_J]$ (M is the number of words in x_i). The attribution scores for this word on embedding column k is R^c_{ijk}. The attribution score for this word is $\sum_k R^c_{ijk}$. The unweighted document representation for x_i is:

$$1/J \sum_j v_j$$

The weighted document representation for x_i is:

$$1/J \sum_j \left(\sum_k R^c_{ijk} \right) v_j$$

The document representation weighted on each individual embedded feature is:

$$1/J \sum_j \left(\sum_k R^c_{ijk} v_j \right)$$

One of the metric examining whether an explanation is useful is to determine whether the truly important features are assigned high attribution scores. Therefore, the weighted document representation should be able to represent the corresponding true class better than unweighted representation, as the words that are important to the prediction are more "influential" in the document representation.

To evaluate whether the weighted documents representations are more representative to its classification result, experiments can be devised both qualitatively and quantitatively:

- Visualization the document representations

  The document representations are of 300 dimensions since the word embedding chosen here is of 300 dimensions. To visualize it, we need to reduce the dimensionality of document representations. Principal component analysis (PCA) is a dimension reduction technique that convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables [12]. A PCA projection can be generated for all documents. From the visualization, a qualitative observation can be made whether the documents from different classes are separated into different clusters.

- Classification on document representations

  Classification task on the document representations is set up with common models such as SVM, K-nearest neighbors or random forest. Overall higher accuracy is expected for weighted document representations if the truly important features are assigned more weights.

4.3 Embedded Feature Removing

Though there are drawbacks to deleting words evaluation method, it is hard to deny how effective and intuitive the method is. Therefore, a new evaluation method is proposed to evaluate whether

the truly important features are assigned high attribution scores by performing perturbations on embedded features.

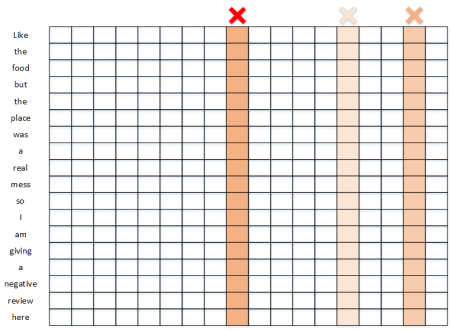


Figure 4.1: Embedded feature removing

First, attribution scores on the embedded features should be generated. The embedded document can be seen as a matrix. For a document x_i of class c, the attribution score for word j on column k is denoted as R^c_{ijk}. The attribution score for this word is calculated by $\sum_k R^c_{ijk}$. Here, an embedding column is considered a feature instead of a word. Therefore, the attribution score for a embedding column is calculated by $\sum_j R^c_{ijk}$. The embedding columns can be removed by setting all values in the corresponding columns to be 0.

The following evaluation can thus be conducted:

1. Remove the embedding columns with the largest attribution scores

   The embedding columns with the largest attribution scores for the true class are removed. For LRP, the predicted probability for this class will be lower. The accuracy is expected to be lower. For saliency map, the features with the largest influence (both positive and negative) on the output will be removed. The purpose of this evaluation is to determine whether the important features are assigned high attribution scores.

2. Remove embedding columns with the smallest attribution scores

   The embedding columns with the smallest absolute attribution scores for the true class are the ones to be removed. For both methods, the predicted probability should not be affected more than randomly removing an embedding column. The purpose of this evaluation is to assess whether features with low attribution scores are truly unimportant features.

3. Remove the embedding columns that contribute differently for different classes

   For a document x_i, the attribution difference between true class c and class c' for embedding column k is $\sum_j (R^c_{ijk} - R^{c'}_{ijk})$. For LRP, when the columns with the largest attribution differences are removed, the predicted probability for class c will decrease while the probability for class c' should increase. For saliency map, since the largest attribution can be both positive and negative, the results are hard to predict theoretically.

4.4 Convolutional Filter Removing

In this section, the attribution scores on the convolution layers are evaluated. For TextCNN, the convolution window mimics an n-gram feature. In a TextCNN, a filter (neuron/kernel) is a set of weights for the vectors of a certain number of words. The output of a filter is the higher representation for these words. In a filter, only the convolution window with the maximum value

has an impact on the output. Thus, removing a filter on the convolutional layers is equivalent to removing higher representations of an n-gram feature.

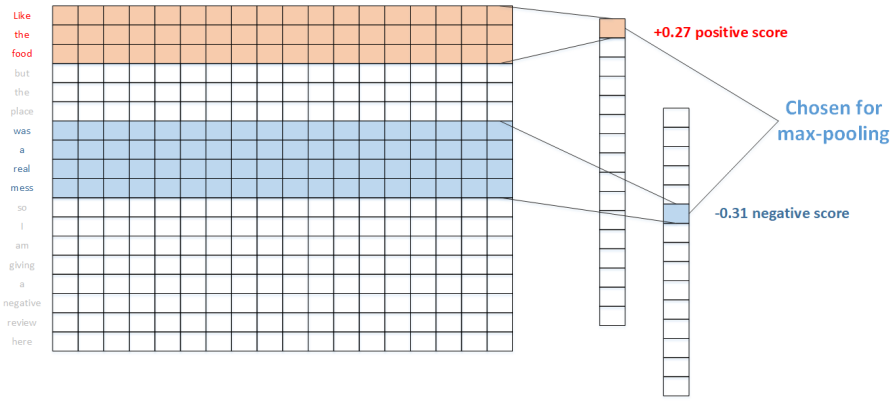


Figure 4.2: Convolutional filter removing

A filter is considered a feature. For a filter, at most one non-zero attribution score will be assigned, which represents the subset attribution scores of the corresponding words. For example, in Figure 4.2, when the filter that represents "was a real mess" is removed, the impact of "a real mess" or "was a real" might still exist.

The evaluations can be conducted similarly to 4.1:

1. Remove filters with the largest attribution scores

   When the filters with the largest attribution scores for the true class are removed, the impact of n-gram features with the largest attribution scores are removed. For LRP, the predicted probability for the true class will decrease. The accuracy is expected to be lower. For saliency map, it is hard to predict the results.

2. Remove filters with the smallest attribution scores

   The filters with the smallest absolute scores are to be removed. For both methods, the model accuracy is expected to decrease slower than randomly removing filters.

3. Remove filters that contribute differently for different classes

   A filter i is assigned attribution score R_i^c for true class c. The attribution difference for filter i is $(R_i^c - R_i^{c'})$. For LRP, when the filters with the largest attribution differeces are removed, the predicted probability for class c will decrease while the probability for class c' should increase.

Chapter 5

Interactive Visualization Demo

In this section, details of the visualization application with the "US consumer finance complaint dataset" will be introduced. This application is developed as a demo to show the possibility of how attribution scores can be used in business.

1. **Overview**

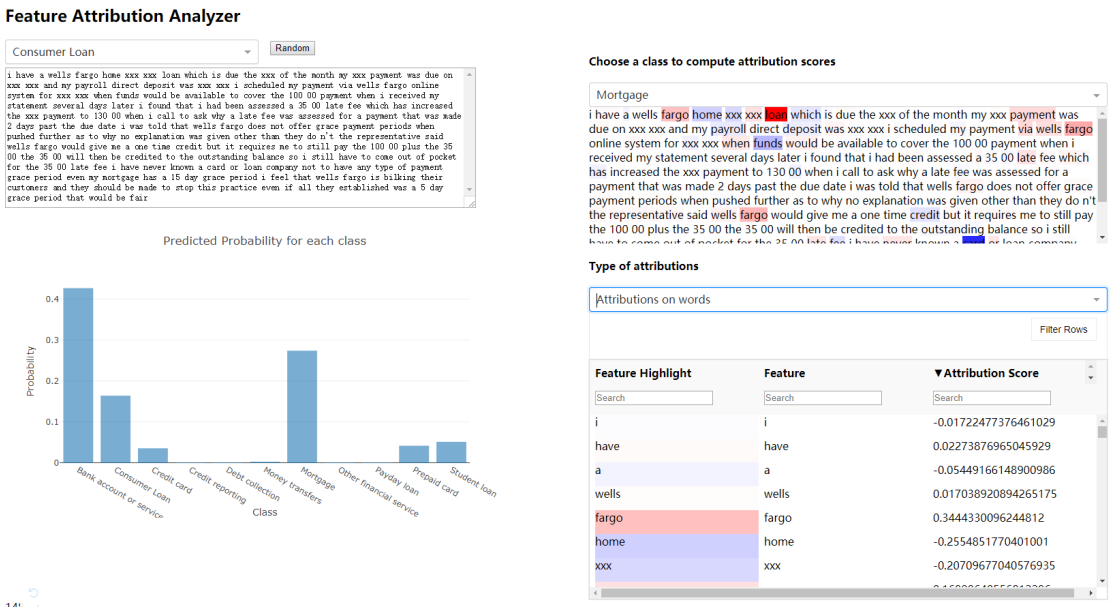


Figure 5.1: Screen shot of visualization demo

The visualization can be roughly divided into prediction visualization and explanation visualization.

2. **Prediction visualization**

   Based on a chosen class, a random document belonging to this class can be selected. The predicted probabilities for each class will be plotted. From the plot, it can be determined whether this document is correctly classified by the model.

3. **Explanation visualization**

When visualizing the explanations, it is possible to generate attribution scores based on a chosen class. In addition, there are four different types of explanations that can be chosen: “attributions on words“, “attributions on ngram“, “attribution differences on words“ and “attribution differences on ngram“. Type of attributions can also be specified.

- Explanation on words

  Explanation of words is visualized in two different ways. Firstly, the attribution scores on each word are highlighted and visualized in the text. Secondly, a table can be generated to list the words and their attribution values.

- Explanation on n-gram features

  To make use of attribution-based explanations in the real world, a more in-depth understanding can be established by assessing the importance of n-gram features in addition to word features.

  From section 4.2, based on the architecture of TextCNN, the attribution score on a filter represents part of the attribution scores for an n-gram feature. Specifically speaking, a filter of size 3 represents a tri-gram feature.

  The n-gram features and their corresponding attribution scores are organized in the table.

- Identify “weak“ features

  The definition of a “weak“ feature is a feature that contributes more positively to a wrong class instead of the true class. By examining only the attribution scores based on the true class, the reasons why a document could also belong to another class is not always clear. To resolve this problem, the attribution differences are calculated, which is the attribution differences for each word/n-gram features between the chosen class and the true class. The features and their corresponding attribution differences are organized in the table.

Chapter 6

Experiments

In this chapter, we will give a detailed description of the datasets used in the experiments. Then, the training results on each of the dataset will be presented as well. In our evaluation experiments on attribution scores, we will first introduce the evaluation on word level with document representations both qualitatively and quantitatively. Then, we will describe the evaluation of embedded-document level and discuss the results. Next, we will introduce the evaluation of n-gram level with analysis on the results. Finally, with the help of our visualization tool, we will demonstrate the analysis on a misclassification example.

6.1 Data Description

6.1.1 Introduction

1. Yelp review dataset

   The dataset is a collection of customer reviews on Yelp. For every review text, the customer gave them a rating. Also, other customers commented about whether a review is cool/funny/useful by giving upvotes.

business_id	ID of the business being reviewed
date	Day the review was posted
review_id	ID for the posted review
stars	15 rating for the business
text	Review text
user_id	User's id
{cool, funny, useful}	Comments on the review, number of upvotes from other users

Table 6.1: Yelp review dataset overview

2. US consumer finance complaints

   The dataset is a collection of customer complaints about financial products that contains 11 types. Each complaint contains one or more sentences. The original dataset contains 18 columns including "date_received", "product", "sub_product", "issue", "sub_issue", "consumer_complaint_narrative", "company_public_response", "company", "state", "zipcode",

“tags“, “consumer_consent_provided“, “submitted_via“, “date_sent_to_company“, “company_response_to_consumer“, “timely_response“, “consumer_disputed¿‘ and “complaint_id“.

To establish a classification task, only “product“ and “consumer_complaint_narrative“ are needed. “product“ denotes which financial product the complaint is about, which will be the label. “consumer_complaint_narrative“ is the customer’s complaint.

6.1.2 Data Transformation

1. Yelp review dataset

   Data tranformation steps:

   a. Extract only “text“, “stars“ and “funny“

   b. Remove missing data

   c. Remove short documents with less than 100 characters

   d. Remove neutral reviews with three stars

   e. Remove reviews with only one upvote in “funny“ as the upvotes are not reliable

   An overview of the transformed data is shown below.

	stars	text	funny
0	1	This is an excellent place for children and fa...	0
1	0	Food was good but nothing special and sides a ...	0
2	0	only one star as I had to. Manager is awful, l...	0
3	0	Just celebrated my 17th wedding anniversary wi...	0
4	1	You know Downtown has changed a lot and with i...	0

Figure 6.1: Yelp transformed data overview

After data transformation, the Yelp dataset contains 55,790 documents. The number of documents in each class is plotted below.

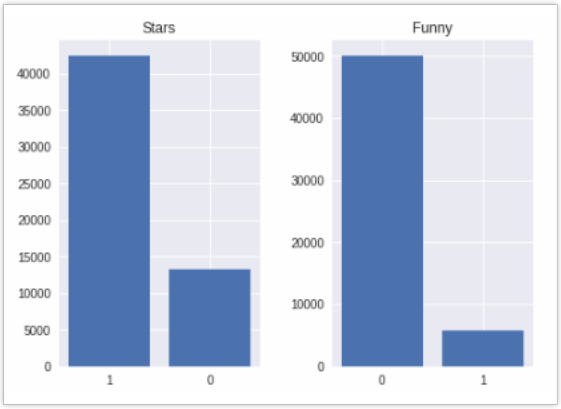


Figure 6.2: Number of documents in each class for “stars“ and “funny“

The average length of the documents is 22. The longest and shortest length of the documents is 208 and 6.

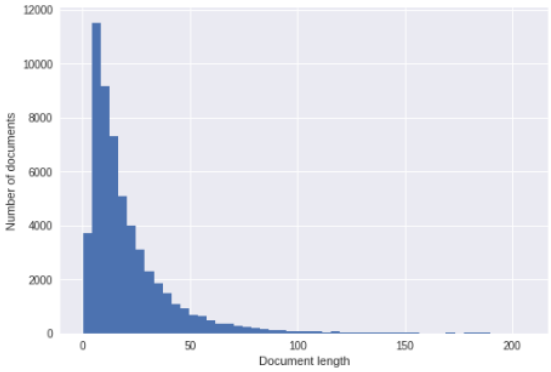


Figure 6.3: Yelp data word count distribution

2. US consumer finance complaints

   Data transformation steps:

   a. Extract only "product" and "consumer_complaint_narrative"

   b. Remove missing data

   c. Remove short documents with less than 100 characters

   Figure 6.4 shows the data overview after transformations.

	product	consumer_complaint_narrative
0	Credit reporting	On my credit report there is a line item under...
1	Student loan	I recently fell behind on my student loan paym...
2	Bank account or service	I am a retired XXXX and I had my assets in a m...
3	Bank account or service	I opened an account with Bank of America in XX...
4	Debt collection	An apartment complex turned a debt over to Alp...

Figure 6.4: Complaint data overview

The dataset contains 64,821 labelled documents after data transformation. The numbers of documents in each class are imbalanced as shown in Figure 6.5.

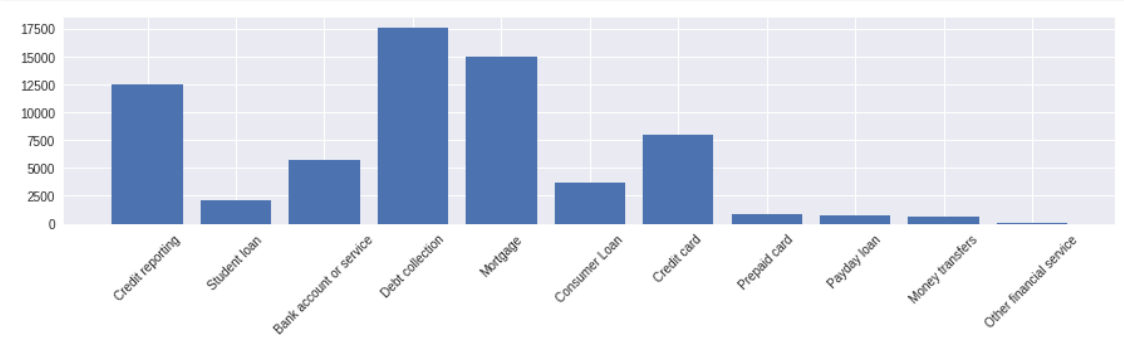


Figure 6.5: Number of documents in each class in complaint data

The average length (word/symbol count) of each document is 198. The longest and shortest length of the documents is 912 and 13.

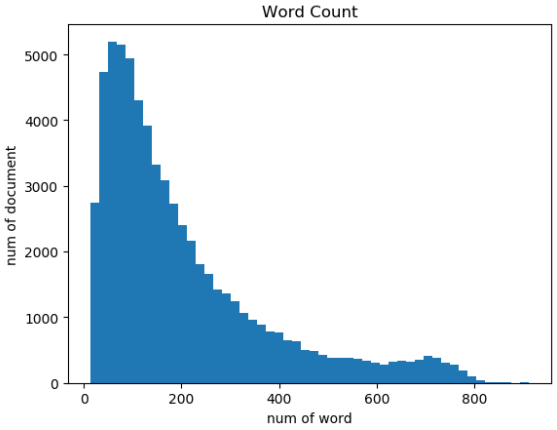


Figure 6.6: Complaint data word count distribution

6.1.3 Pre-processing

For both Yelp review dataset and Consumer complaint dataset, the preprocessing steps are very similar.

a) *Tokenization.* The tokenizer from "keras" was used here to vectorize the raw documents into lists of token (word) indexes. The total number of tokens is limited based on tf-idf. The punctuation is removed by default.

b) *Embedding matrix.* For both datasets, an embedding matrix is built up as the parameters in the embedding layer in neural networks. GloVe embedding of dimension 300 is used here [1].

c) *Padding.* The sequences into the neural network should be of the same length. Therefore, all sequences are either padded with zeros or cut down to a fixed length.

d) *Data split.* Before training, both datasets are split into training and validation sets in a stratified way so that the distribution of classes stays the same in both sets. The training set accounts for 80% of the entire dataset.

	Yelp review	US consumer finance complaint
Number of tokens	15000	20000
Length of sequences	200	300
Embedding dimension	300	300

Table 6.2: "Parameters in pre-processing"

6.2 Model Training

6.2.1 Training on Yelp Review Data

1. Model architecture

[1] http://nlp.stanford.edu/data/glove.6B.zip

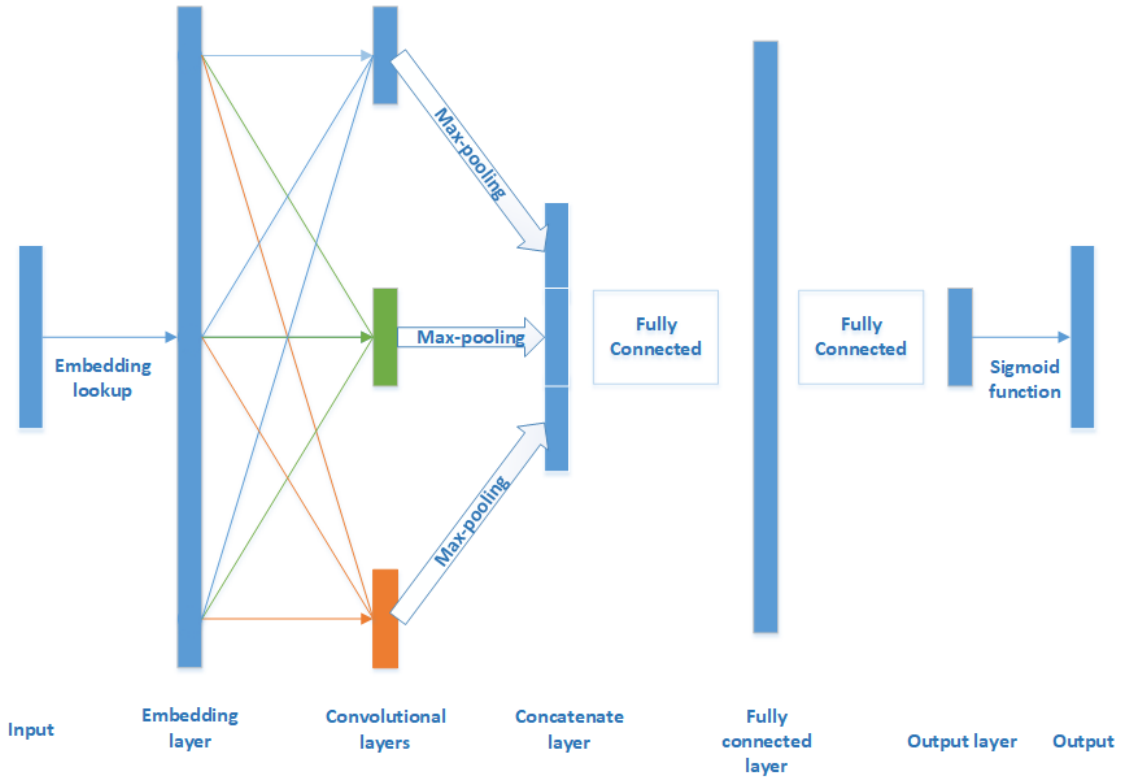


Figure 6.7: Model architecture for Yelp review dataset

Since the tasks on the Yelp review dataset are all binary classifications, the output layer is of size 1. The sigmoid function is used as activation function for the output layer. The same setting is used for all tasks.

Based on a typical TextCNN architecture, a fully connected layer of size 256 is added before the output layer to prevent overfitting. For the same purpose, dropout is added after the "Concatenate layer" and the "Fully connected layer". Also, l2-regularization is added to the output layer.

2. Hyper-parameters

Table 6.3: Hyper-parameters

filter sizes	nr. of filters	dropout	batch size	l2-regularization	optimizer	loss
[3, 4, 5]	32	0.2	128	0.03	adam	binary_crossentropy

From Figure 6.2, it is easy to observe that the data is severely imbalanced for all four tasks. Therefore, class weights are computed before training the model to eliminate the impact of data imbalance.

3. Training results

Table 6.4: Training results on Yelp review dataset

task	epochs	loss	accuracy	validation loss	validation accuracy
“stars“	3	0.0788	97.6%	0.1375	95.27%
“funny“	2	0.2675	90.34%	0.2717	90.33%

6.2.2 Training on US Consumer Finance Complaint Data

1. Model architecture

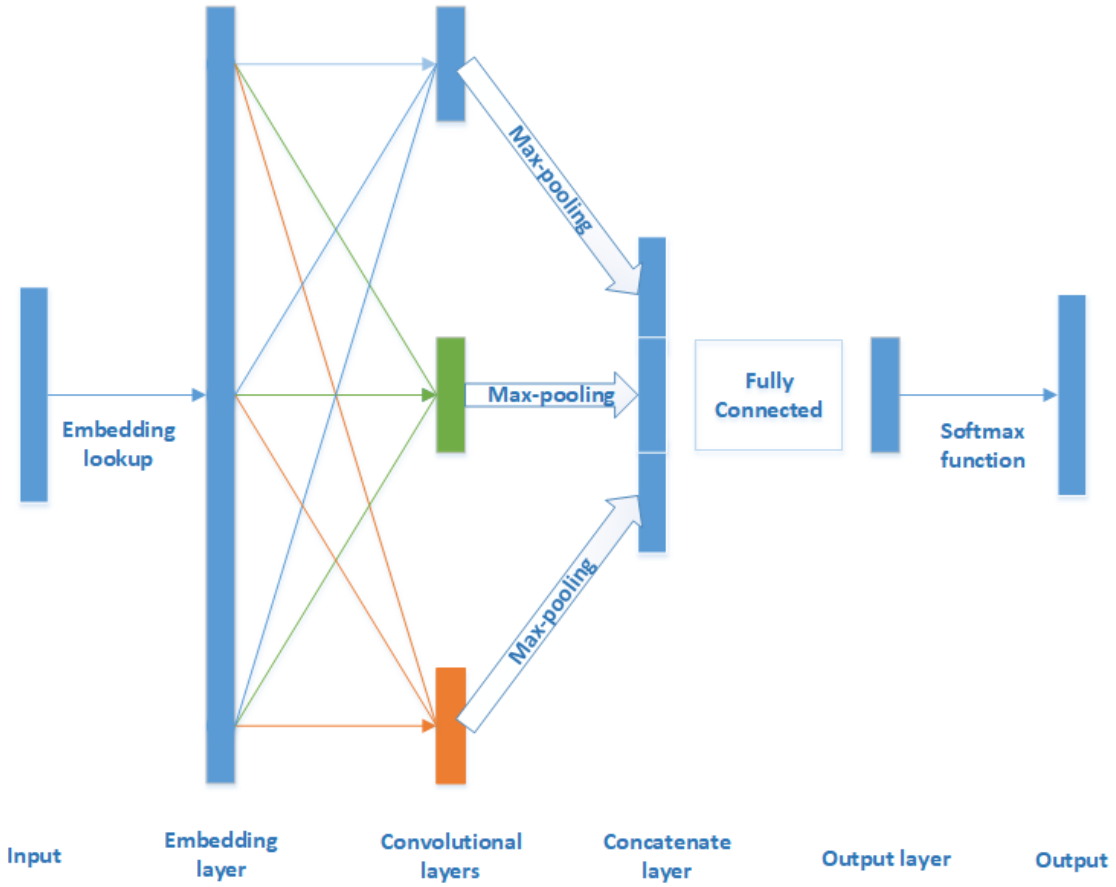


Figure 6.8: Model architecture for US consumer finance complaint dataset

The task is to predict about which product the consumer is complaining about, which is a multi-class classification problem. Therefore, the output layer size is 11, which is the number of classes. Softmax function is used as the activation function for the output layer.

A typical TextCNN architecture is used here. To prevent overfitting, dropout was added after “Concatenate layer“. Regularization is performed on the output layer.

2. Hyper-parameters

Table 6.5: Hyper-parameters

filter sizes	nr. of filters	dropout	batch size	l2-regularization	optimizer	loss
[3, 4, 5]	64	0.5	64	0.03	adam	categorical_crossentropy

From Figure 6.5, the number of documents in each class is different. To eliminate the impact of data imbalance, class weights are applied when fitting the model.

3. Training results

Table 6.6: Training results

epochs	loss	accuracy	validation loss	validation accuracy
15	0.3314	89.5%	0.6031	85.45%

6.3 Document Representations

In the experiments in this section, five different document representations are generated for each document:

- Unweighted representation: the mean of vector representations of all words in the document
- Weighted representation by LRP: the mean of weighted vector representations. The attribution scores generated by LRP on each word are used as weights
- Weighted individually representation by LRP: the mean of weighted vector representations. The attribution scores generated by LRP on each individual embedded features are used as weights
- Weighted representation by saliency map: the mean of weighted vector representations. The attribution scores generated by saliency map on each word are used as weights
- Weighted individually representation by saliency map: the mean of weighted vector representations. The attribution scores generated by saliency map on each individual embedded features are used as weights

For classification tasks on the document representations, the models used are listed in the table below. The parameters not mentioned are all set to default according to the scikit-learn package.

Table 6.7: Classification models and parameters

Model	Parameters	Model	Parameters
KNeighbors	n_neighbors = 10	Ada Boost	default
SVC	kernel = “rbf“, C = 0.025	Gradient Boosting	default
NuSVC	default	Gaussian Naive Bayes	default
Decision Tree	default	Linear Discriminant Analysis	default
Random Forest	default	Quadratic Discriminant Analysis	default

6.3.1 Experiments on Yelp Review Dataset on Task "stars"

For this experiments, 2000 correctly classified documents from both classes are selected (1000 documents per class) for task "stars". Different types of document representations are generated for all documents.

PCA Projection

To qualitatively assess how well the attribution scores represent the model predictions, A PCA is conducted to project the documents into 2-dimensional space. Each document is plotted and labelled according to its true class.

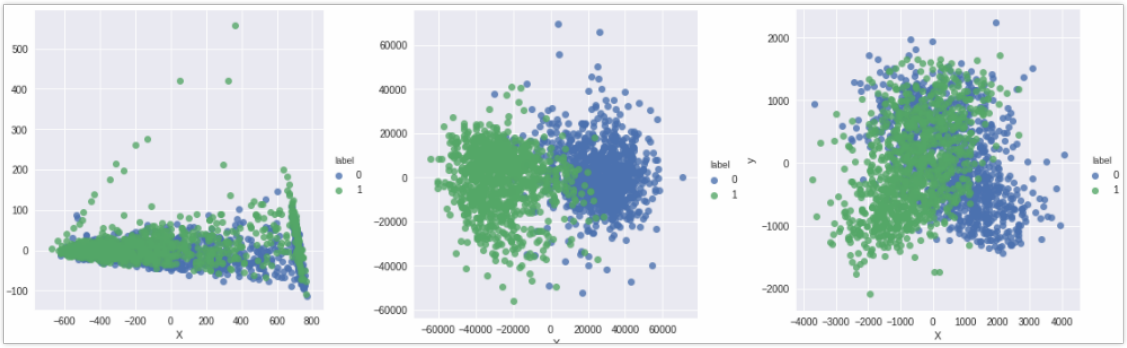


Figure 6.9: PCA projections when using unweighted representation (left), weighted representation by LRP (mid) and weighted individually representation by LRP (right) on label "stars"

In the PCA projection for unweighted representations, no obvious clusters can be observed. In the weighted representations, the documents are separated into two clusters. Though the clusters still overlap with each other. In the weighted individually representations, the clusters are not separated.

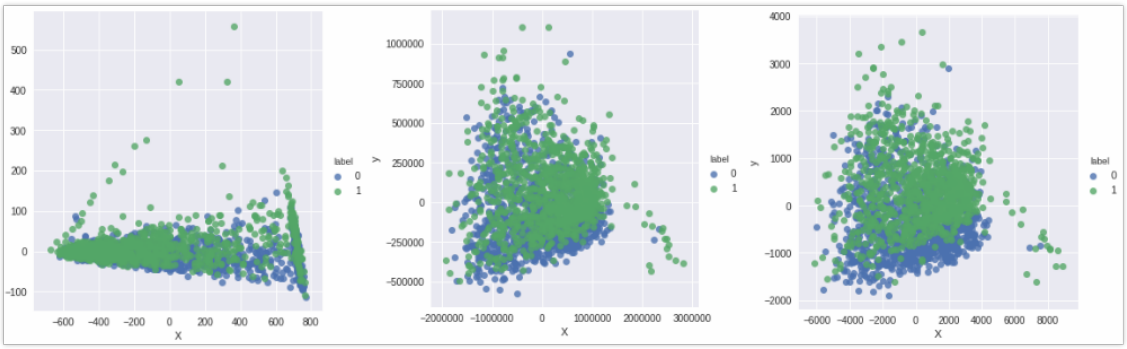


Figure 6.10: PCA projections when using unweighted representation (left), weighted representation by saliency (mid) and weighted individually representation by saliency (right)

When using the attribution scores generated by saliency map, the separation of each class is not very obvious compared to using LRP attribution scores.

Classify Document Representations

In the classification tasks, each type of document representations is split into training data (80%) and test data (20%). Ten different models are trained on the training data. The accuracy and losses are calculated on the test data.

To compare the results for all types of document representations, the maximum accuracy, mean accuracy, mean log loss and minimum log loss are calculated based on the results from all ten models.

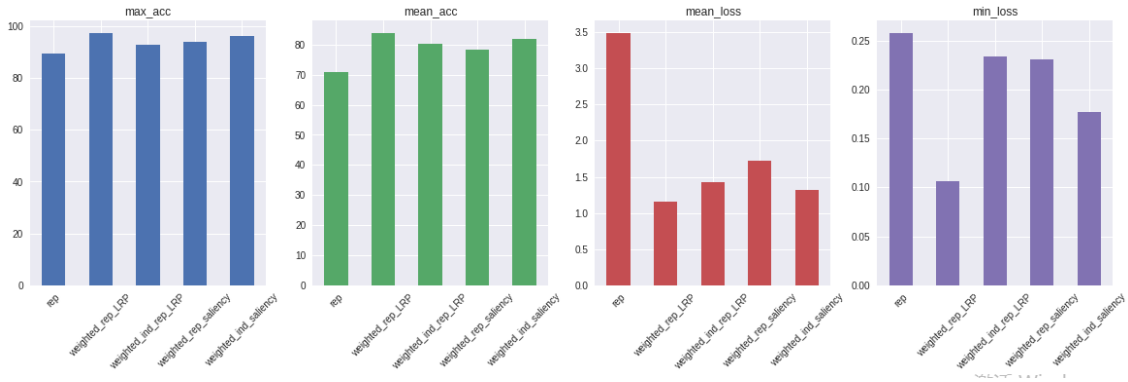


Figure 6.11: Classification accuracy and loss on all types of document representations

From the comparison, the weighted document representations always achieve better performance. The weighted document representations according to LRP seems to have received the best classification results.

6.3.2 Experiments on Yelp Review Dataset on Task "funny"

For this experiments, 2000 correctly classified documents from both classes are selected (1000 documents per class) for the task "funny". Different types of document representations are generated for all documents.

PCA Projection

To qualitatively assess how well the attribution scores represent the model predictions, A PCA is conducted to project the documents into 2-dimensional space. Each document is plotted and labelled according to its true class.

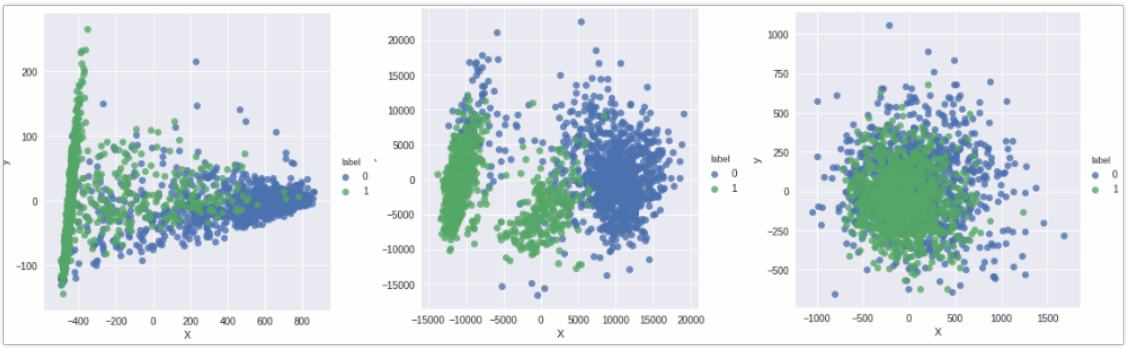


Figure 6.12: PCA projections when using unweighted representation (left), weighted representation by LRP (mid) and weighted individually representation by LRP (right) on label "funny"

In the PCA projection for unweighted representations, the documents from two classes severely overlap with each other. In the weighted representations, the documents are separated into a few clusters. Though the clusters still overlap with each other. In the weighted individually representations, the clusters are not separated.

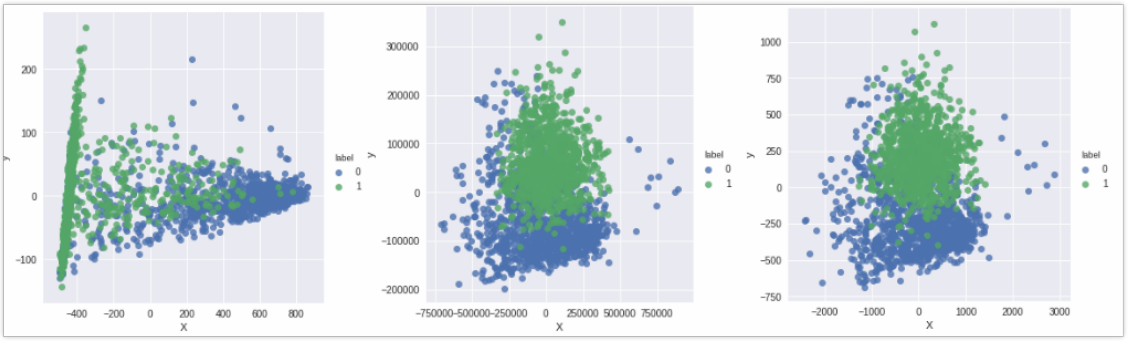


Figure 6.13: PCA projections when using unweighted representation (left), weighted representation by saliency (mid) and weighted individually representation by saliency (right) on label "funny"

When using the attribution scores generated by saliency map, the separation of each class is observable compared to the unweighted document representation.

Classify Document Representations

In the classification tasks, each type of document representations is split into training data (80%) and test data (20%). Ten different models are trained on the training data. The accuracy and losses are calculated on the test data.

To compare the results for all types of document representations, the maximum accuracy, mean accuracy, mean log loss and minimum log loss are calculated based on the results from all ten models.

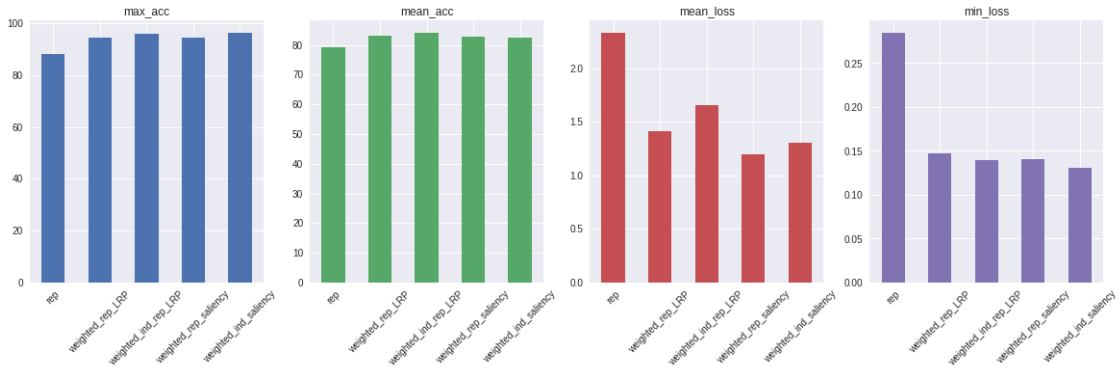


Figure 6.14: Classification accuracy and loss on all types of document representations

From the comparison, the weighted document representations always achieve better performance, especially by observing the losses. It is hard to determine which type of document representation has received the best classification results.

6.3.3 Experiments on US Consumer Finance Complaint Dataset

For the experiments, 2000 correctly classified documents from 10 classes (200 documents per class) are selected (class 'Other financial service' is not included due to insufficient data in the test set). Document representations are generated for all the documents.

PCA Projection

A PCA is conducted to project the documents into 2-dimensional space. Each document is plotted and colored according to its class.

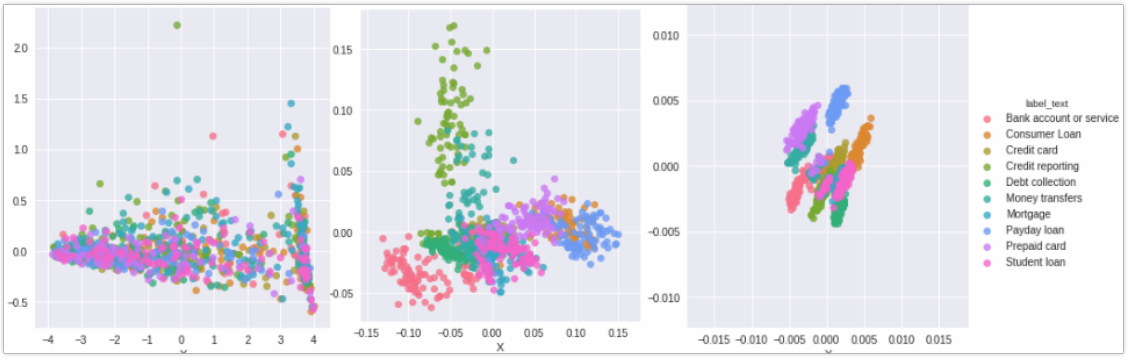


Figure 6.15: PCA projections when using unweighted representation (left), weighted representation by LRP (mid) and weighted individually representation by LRP (right)

In the PCA projection for unweighted representations, no obvious clusters can be observed. In the weighted representations, the documents are separated into a few clusters. Though the clusters still overlap with each other. In the weighted individually representations, the clusters are more concentrated and easy to observe.

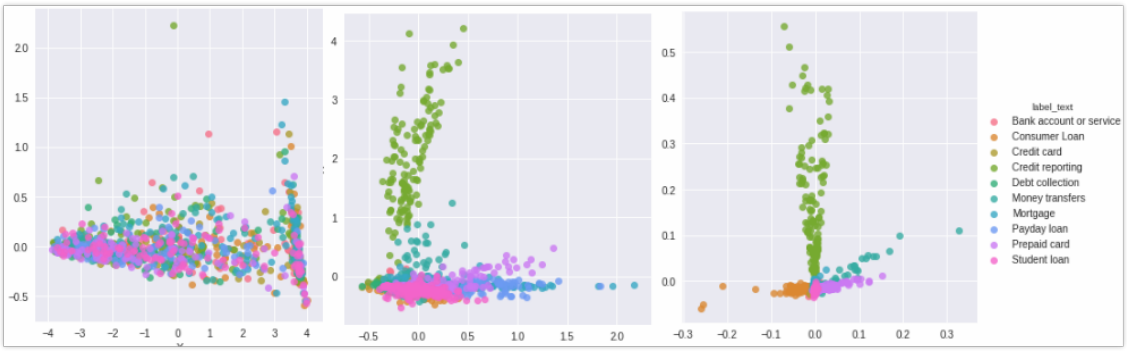


Figure 6.16: PCA projections when using unweighted representation (left), weighted representation by saliency (mid) and weighted individually representation by saliency (right)

When using the attribution scores generated by saliency map as weights, the results show some similarities compared to LRP attribution scores. Compared to unweighted representations, the separation of each class for weighted representations is shown. In the weighted individually representations, the clusters are more concentrated. However, The clusters are not as evident as projection using LRP attributions.

Classify Document Representations

In the classification tasks, each type of document representations is split into training data (80%) and test data (20%). Ten different models are trained on the training data. The accuracy and losses are all calculated on the test data.

To compared the results for all types of document representations, the maximum accuracy, mean accuracy, mean log loss and minimum log loss are calculated based on the results from all ten models.

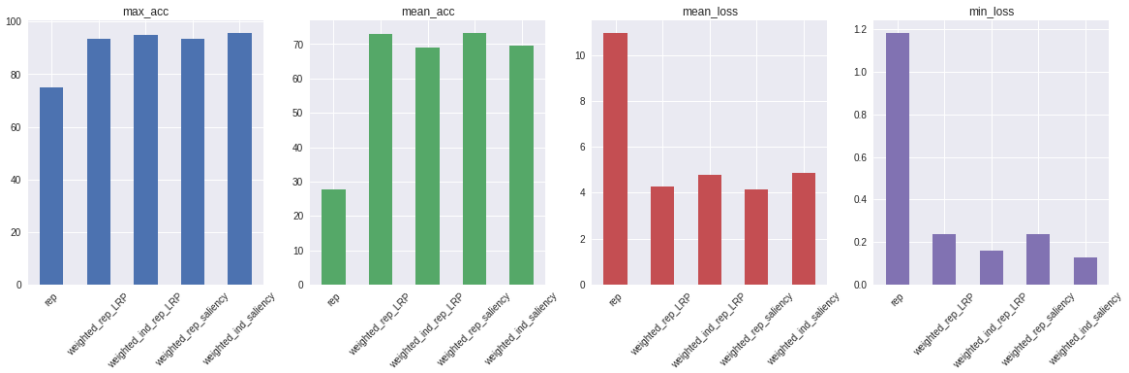


Figure 6.17: Classification accuracy and loss on all types of document representations

From the comparison, the weighted document representations always achieve better performance. For LRP and saliency map, the representation weighted on word both resulted in better mean accuracy and loss. But the maximum accuracy was lower than using representations weighted individually on embedded features.

6.3.4 Conclusions

In accordance with our research question 3, the evaluation was conducted without making any changes to the inputs and the model. Document representations were generated with attribution scores on each word as weights. Compare to the unweighted document representations, clearer clusters for each label were observed with the weighted representations. Moreover, when using weights, better classification results were achieved on both datasets.

Towards research question 4, a) the evaluation was able to produce comparable results in the PCA projection and measurable results in document representation classification. b) The evaluation also answers whether the truly important features can be found. By amplifying the influence of features with higher attribution scores, both PCA projection and classification results support the conclusion that saliency map and LRP found features that are relevant to the documents' true labels.

6.4 Feature Removing Experiments with Embedded Documents

In this section, columns of the embedded document (pre-trained word embedding) will be considered the features of each document. The attribution scores for a document for a class are the attribution scores resulting from the predicted value of this particular class before the "softmax"/"sigmoid" function.

The features with the smallest/largest attribution scores based on the validation set will be removed. The model's performance on correctly classified documents from a particular class in the validation set is then measured in several aspects. First, the model's accuracy loss is measured. Secondly, the number of misclassifications in each class is recorded.

6.4.1 Experiments on Yelp Review Dataset

Experiments on Task "stars"

1. Attribution scores

   Before the evaluation experiments, some exploratory visualizations are made to gain insights into the generated attribution scores on the word embedding layer.

   Attribution scores on words: Attribution scores on text data are usually visualized and analyzed in words. Below is a heat map visualization of the attributions of each word. The document is correctly classified as stars "1". The above and below text shows the attribution scores generated by LRP and saliency map. For the attributions generated by LRP, red represents a positive value while blue represents a negative value. The darker the color, the larger the absolute value. The attribution scores generated by saliency map are all positive values while the attribution scores generated by LRP distinguishes between positive and negative impacts.

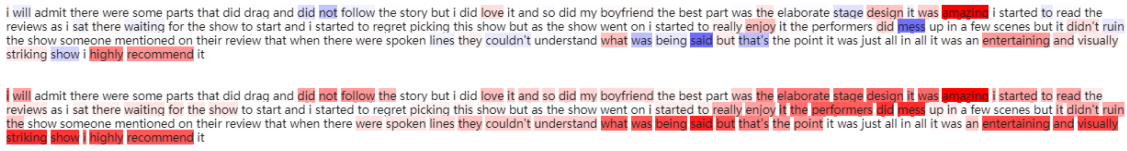


Figure 6.18: Examples of attribution scores visualized on text

Attribution scores on embedded document The attribution score of a word is the sum of a vector of attribution scores. To investigate further, a sample heat map of attribution scores visualized on the embedded document. The figure shows part of a document correctly classified as stars "1".

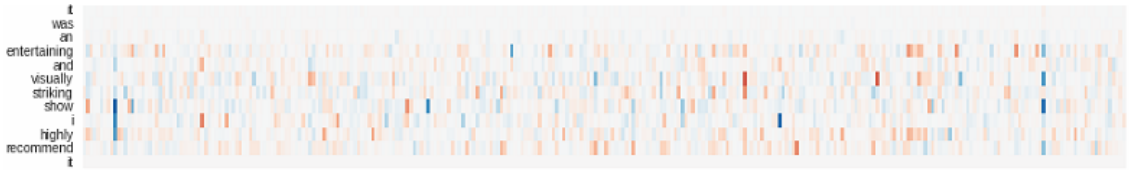


Figure 6.19: Attribution scores (by LRP) heatmap on embedded document

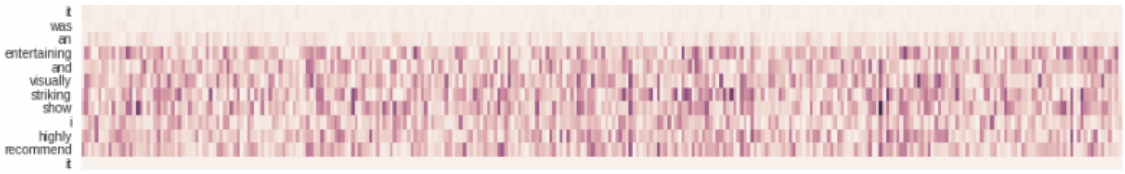


Figure 6.20: Attribution scores (by Saliency map) heatmap on embedded document

In the above figures, apart from the differences in signs, similarities can be observed. In some columns, the attribution scores from different rows are similar. To observe the similarities, the attribution scores columns are sorted by the sum of attribution scores on this particular document.

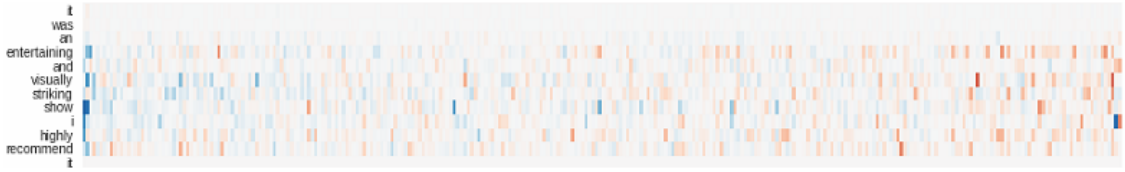


Figure 6.21: Sorted attribution scores (by LRP) heatmap on word embedding

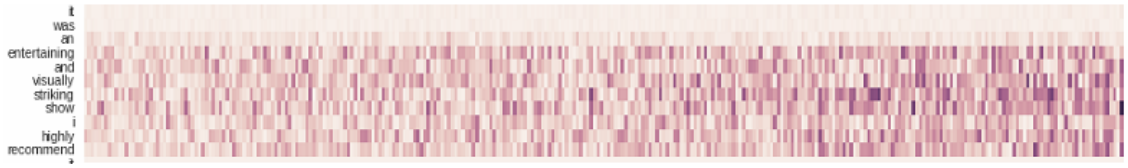


Figure 6.22: Sorted attribution scores (by Saliency map) heatmap on word embedding

Attribution score sums by the column of multiple documents: From the above figures, negative values and positive values are roughly separated in the heatmap of LRP attribution scores. The colors on the left and right sides are darker. In the heatmap of saliency map attribution scores, the colors on the right side are mostly darker than the left side. This observation can be evidence that for a certain outcome, different columns of word embedding might have a different impact, and that attributions from the same columns show a certain level of consistency.

Experiments are designed to visualize the sum of attribution scores of embedded documents by column. Collections of 1,000 documents are randomly selected from classes "0" and "1".

All the documents are correctly classified. The attribution score of a document is a row of a vector in the below figures. The left figures are the original attribution score sum while the right side figures are the sorted scores based on the column sum of attribution score sum.

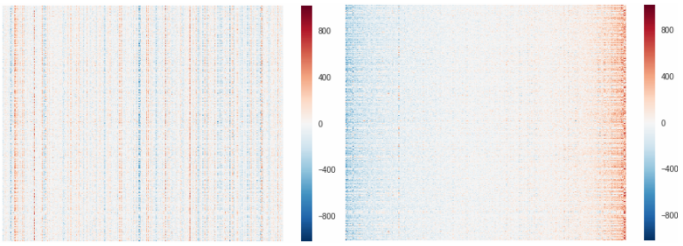


Figure 6.23: Heatmap of attribution score (LRP) sum for “0“ stars embedded documents

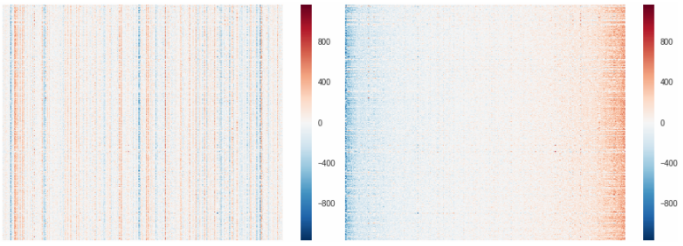


Figure 6.24: Heatmap of attribution score (LRP) sum for “1“ stars embedded documents

For different classed, the attribution score (LRP) sum shows some differences in column values. However, by inspecting the sorted attribution score sum, in the bottom 10 and top 10 columns, six are the same for both classes. The similar but a lot less obvious observation can be made using attribution scores generated by saliency map.

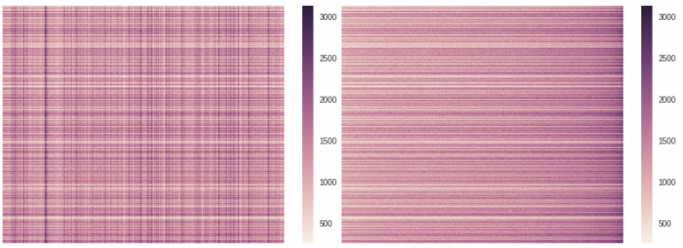


Figure 6.25: Heatmap of attribution score (saliency map) sum for “0“ stars embedded documents

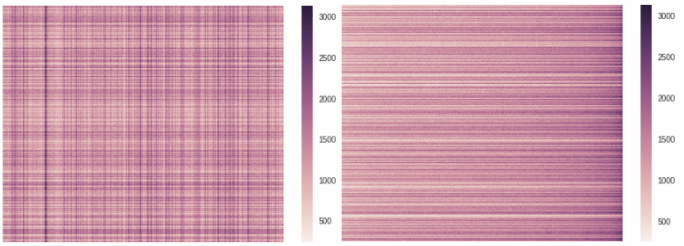


Figure 6.26: Heatmap of attribution score (saliency map) sum for “1“ stars embedded documents

2. Evaluate the attribution scores on embedding columns

Three types of experiments are conducted: by removing both the most important positive columns (columns with the largest attribution scores), the most important negative columns (columns with the smallest attribution scores in LRP and largest attribution scores in saliency map) and the least important columns (columns with the smallest absolute attribution scores), the model accuracy is calculated. As mentioned in the above experiment, only documents that are classified correctly are used to calculate attribution values or model prediction.

In this experiment, the original dimension of the word embedding is 300. Every time ten columns are removed, the model accuracy is measured.

- When removing the most important positive columns, the columns with the largest attribution scores will be removed. Documents labelled "1" are used here to compute attribution scores and measure model accuracy. The predicted probability is expected to decrease.
- When removing the most important negative columns, documents labelled "0" are used. Columns with the smallest LRP attribution scores or columns with the largest saliency map attribution scores are removed.
- When removing the least important columns, documents from both classes are used. Columns with the least absolute attribution values are removed.

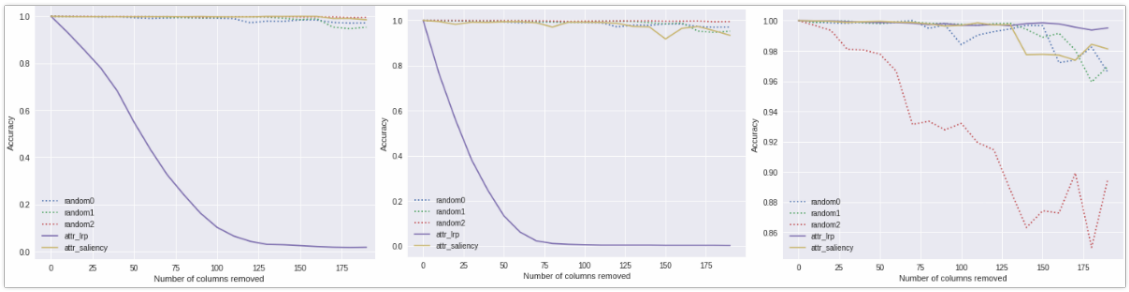


Figure 6.27: Model accuracy when removing embedding columns with the largest (left), smallest (mid) and smallest absolute attributions (right) on task "stars"

From the results from using LRP attribution scores, we can see that after removing 50 largest embedding columns, the accuracy on documents labelled "1" became lower than 60%. After removing 50 smallest embedding columns, the accuracy on documents labelled "0" became less than 20%.

Additional Notes:

3. Determine whether *Assumption 1.* holds

   In this experiment, to avoid computing the LRP attribution scores repeatedly, the order for removing the embedding columns is according to the attribution scores calculated while no features were removed. To determine whether the ranking of attribution scores will change when some embedding columns are removed, the following experiments were conducted.

   The rankings of embedding columns to be removed are calculated in the following ways:

   - Compute the LRP attribution scores. The columns with the larger/smaller sum will be ranked higher.
   - Compute the LRP attribution scores. The column with the largest/smallest sum will be ranked first. After the corresponding column was removed, the attribution score is computed again to get the next column with the largest/smallest attribution value. The ranking is calculated by repeating this process.

On documents correctly classified as "1", the rankings from the largest to the smallest attribution scores on the embedding columns are computed. On documents correctly classified as "0", the opposite rankings are computed for the above two different approaches. Then, the intersections of rankings on the same data are inspected.

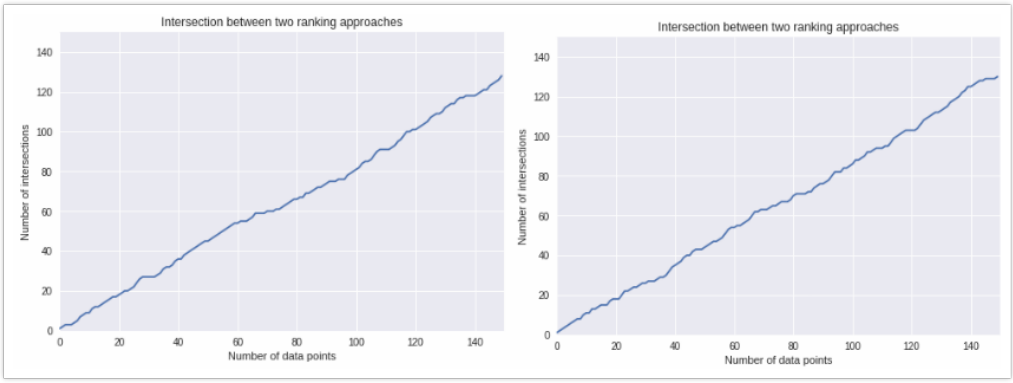


Figure 6.28: Intersections

In the left figure, the rankings are computed on documents labelled "0". Therefore, the rankings are embedding columns with smallest to largest attribution scores. In the right figure, the rankings are computed on documents labelled "1". In both figures, the two different rankings have a large part of intersections. Especially in the top rankings (0-20), the rankings are almost identical. Based on the results, *Assumption 1.* can be applied in our experiments.

4. Validate the correctness of LRP ranking

   From the above results, we can see that when using LRP attributions to remove features (embedding columns) with the largest positive or smallest negative scores, the accuracy plummeted more than other approaches. To validate the correctness of the order to remove features, the following experiment was carried out.

   Perturbations were conducted to the ranking of the features to remove. If the original ranking is correct, the accuracy should decrease more when using the original ranking. 20 features ranked top in the original ranking were chosen. To observe the accuracy differences closely, features were removed one by one.

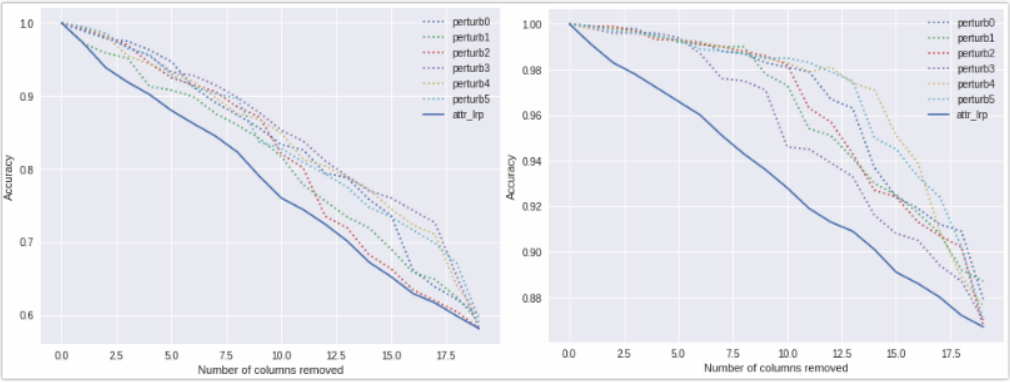


Figure 6.29: Model accuracy when removing embedding columns with the smallest (left) and largest (right) LRP attribution scores

In the left figure, features with negative attributions were removed. The documents labelled "0" were used to compute both attribution scores and accuracy. Documents labelled "1" were used in the right figure where features with positive attributions were removed.

By comparing the results of using the original ranking of the perturbed ranking, the accuracy is always lower when using the original ranking. For perturbed rankings, the accuracy drops quicker after a while. In this experiment, the features with a larger impact on the results were indeed assigned a higher attribution score.

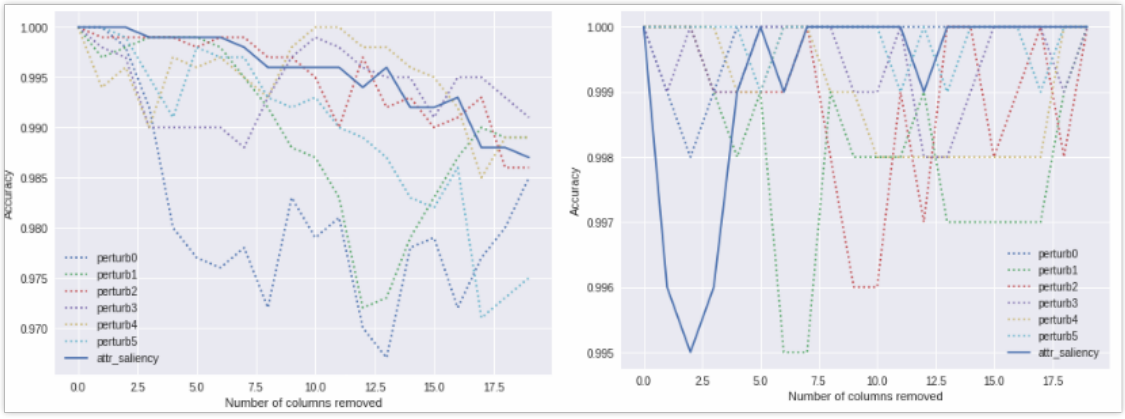


Figure 6.30: Model accuracy when removing embedding columns with the largest (left on documents labeled "0") and largest (right on documents labeled "1") saliency attribution scores

When saliency map is used, the results are different. The original ranking did not stand out among all experiments. The reason is that the attribution scores are not signed. Therefore, the most influential change could be positive or negative. Moreover, due to the non-linearity of the neural network, some changes in the feature could get lost after a few activation functions. Therefore, removing the most "influential" feature does not always result in the largest accuracy change.

5. Visualize the embedding removing on text

Word embedding is a very abstract concept. A column in the embedded document does not have any semantic meaning. Therefore, better visualizations are needed to show the embedding column removing process.

In the above experiment, ten columns are removed each time. To show what attributions are removed, heatmaps of attribution scores are plotted on the text. Every time ten columns are removed, the new attributions will be visualized on a document classified as "1" as an example.

From the top down are visualization as attribution scores on the corresponding embedding columns are removed. When the largest saliency attribution scores are removed, it is clear that the attribution values are indeed a lot smaller. When the smallest saliency attribution scores are removed, the changes are not so dramatic.

Figure 6.31: Attribution scores on text as removing embedding columns with the largest LRP attribution scores

Figure 6.32: Attribution scores on text as removing embedding columns with the smallest LRP attribution scores

Figure 6.33: Attribution scores on text as removing embedding columns with the smallest absolute LRP attribution scores

The visualizations on text when embedding columns are removed based on saliency map are included in the Appendix. When LRP attribution scores are used, the results are different. When the largest LRP attributions are removed, the positive impact becomes smaller or even negative (for word "good"). The negative impacts are not affected much. When removing the smallest LRP attributions, negative impacts become positive (word "good", "recommend" and "wouldn't"). When removing the smallest absolute attributions, the scores did not change as much.

6. Visualizaiton on document representations

   To observe the effects of removing the word embedding columns, we have generated new document representations weighted on each word with attribution scores.

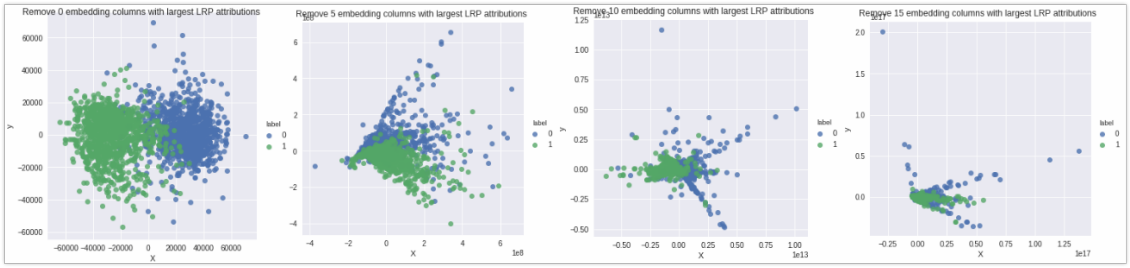


Figure 6.34: Weighted document representations when removing embedding columns based on largest LRP attributions

As the embedding columns with the largest attribution values are removed, it is easy to observe that the documents from different classes are starting to overlap with each other.

Experiments on Task "funny"

The same experiments were conducted on the task "funny". Accuracy was measured each time 10 embedding columns were removed according to the attribution score ranking. The results are shown in the figure below.

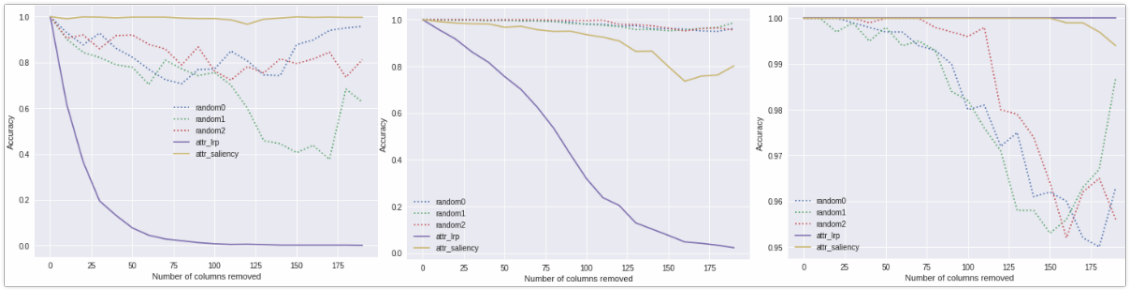


Figure 6.35: Model accuracy when removing embedding columns with the largest (left), smallest (mid) and smallest absolute attributions (right) on task "funny"

From the results from using LRP attribution scores, we can see that after removing 50 largest embedding columns, the accuracy on documents labelled "1" became lower than 20%. After removing 50 smallest embedding columns, the accuracy on documents labelled "0" decreased to around 75%.

Result analysis

For all the experiments on Yelp dataset, by removing the least important embedding columns, experiments with LRP and saliency map approaches have successfully preserved the model accuracy better than random approaches.

When using LRP attribution scores, the model accuracy sees a much more drastic drop when removing embedding columns with the largest or smallest attribution scores. When removing the

embedding columns with positive scores, the predicted value in the model will decrease, some documents labelled “1“ will be predicted “0“, resulting in the accuracy decrease. When removing the embedding columns with negative scores, the predicted value will increase, some documents labelled “0“ will be predicted “1“.

However, when using saliency attribution scores, the accuracy did not decrease much more than randomly removing embedding columns. The reason is that saliency approach cannot distinguish between positive and negative impacts. When removing the most influential features, both positive and negative ones are removed. Thus, the accuracy did not decrease as much as when using LRP attribution scores.

From the above experiments, both LRP and saliency map are able to assign high importance scores to the embedding columns that are important for the model predictions. By using LRP, it appears to be possible to alter the prediction towards one direction.

6.4.2 Experiments on US Consumer Finance Complaint Dataset

In this experiment, attribution scores (for the true class) are also calculated for the embedded documents. To investigate the differences in attribution scores from different classes, the differences between the true class and other classes are calculated as well. The same experiments are repeated based on attribution differences and random attribution scores.

Experiments on documents labelled “bank account or service“

There are 437 documents in the validation set that are correctly classified as “bank account or service“, which is label “0“. Attribution scores are computed for these embedded documents with both saliency map and LRP. To investigate the differences in attribution scores for different classes on the same documents, attribution scores for all classes are computed as well.

The attribution scores are then summed up on the word embedding column axis. The attribution scores dimension is reduced to (437, 300) from (437, 300, 300). Because of the consistency of attribution scores sum for each word embedding column for each document, the attribution scores are further summed up for all documents. The attributions score dimension is then reduced to (300,), which will be the metric to determine which word embedding column as a more significant impact on the model prediction. The same procedure is performed on the attribution scores differences. In this experiment, the word embedding dimension was reduced from 300 to 100 by 10 columns in each step. By setting columns of values in the word embedding, the columns are removed. When removing the word embedding columns, the model accuracy and misclassifications will be recorded.

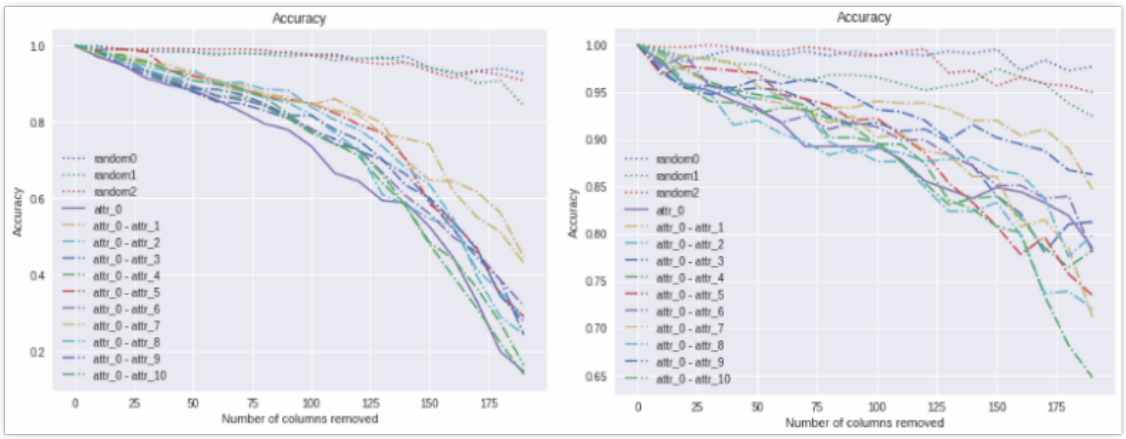


Figure 6.36: The accuracy when removing embedding columns for "bank account or service" documents based on attribution scores by LRP (left) and saliency map (right)

Above is the model accuracy changes when removing the embedding columns with the largest attribution scores or attribution difference scores. From the left figure, the attributions are generated by LRP. Compared to randomly removing the embedding columns, the approaches based on attribution scores or attribution differences have all resulted in a more substantial drop in accuracy. When less than 120 columns are removed, removing based on attribution scores has resulted in lower accuracy than all other approaches.

From the right figure, the attributions are generated by saliency map. The differences between random and attributions can also be observed. However, from the accuracy axis, the accuracy drop here is less than the experiment using LRP. Also, the impact of removing based on attributions is similar to attribution differences. The reason why the accuracy decrease is observed here but not in binary classification tasks is that the attribution scores were calculated for the true class. The positive features account for more than the negative features. Thus, among features with high attribution scores, most of them are positive to the output. It can be observed from the figure that the accuracy usually decreases but seldom increases.

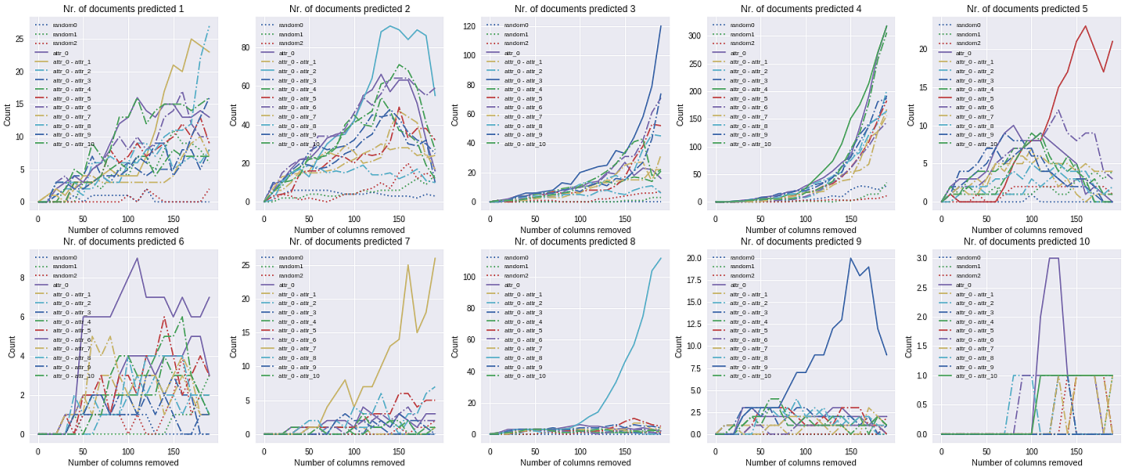


Figure 6.37: The mis-classification count when removing embedding columns for "bank account or service" documents based on attribution scores by **LRP**

When removing the word embedding columns, the model accuracy will drop, which means that

some documents are misclassified. Therefore, every time the word embedding columns are removed, the numbers of documents misclassified to each class are recorded for all approaches. In Figure 6.32, in "Nr. of documents predicted 1", "attr_0 - attr_1" (differences in attribution scores between class 0 and class 1) has a higher number of documents classified as "1" than other approaches. The same trend can be seen in all classes.

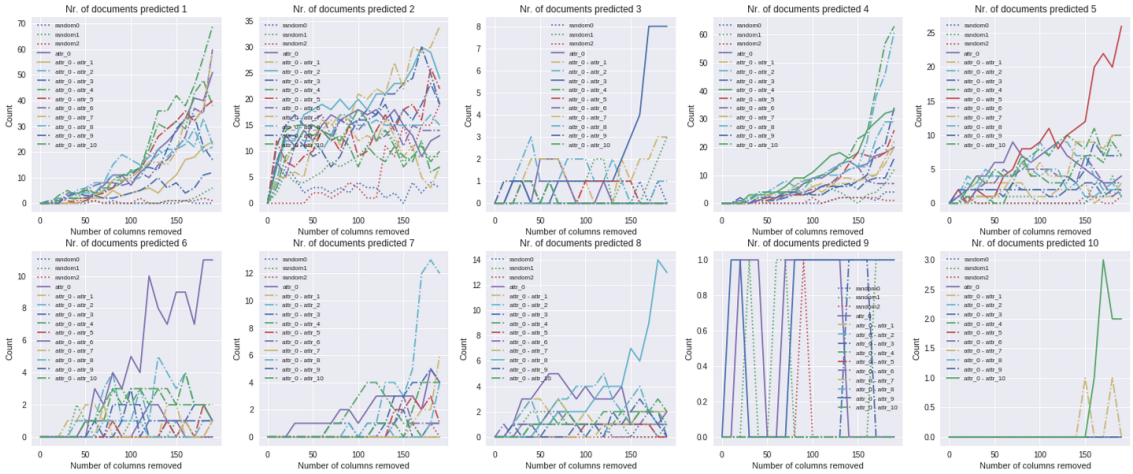


Figure 6.38: The mis-classification count when removing embedding columns for "bank account or service" documents based on attribution scores by **saliency map**

When attribution scores are generated by saliency map, the results are different. In "Nr. of document predicted 6", "attr_0 - attr_6" has resulted in a higher number of documents classified as "6". The same trend is only seen in class "3" and "8". However, there are too few documents classified as "3", "8" and "6", which is not sufficient in jumping to a conclusion.

Experiments on documents labelled "credit card"

The experiments in this section are identical to the experiments on "bank account or service" documents.

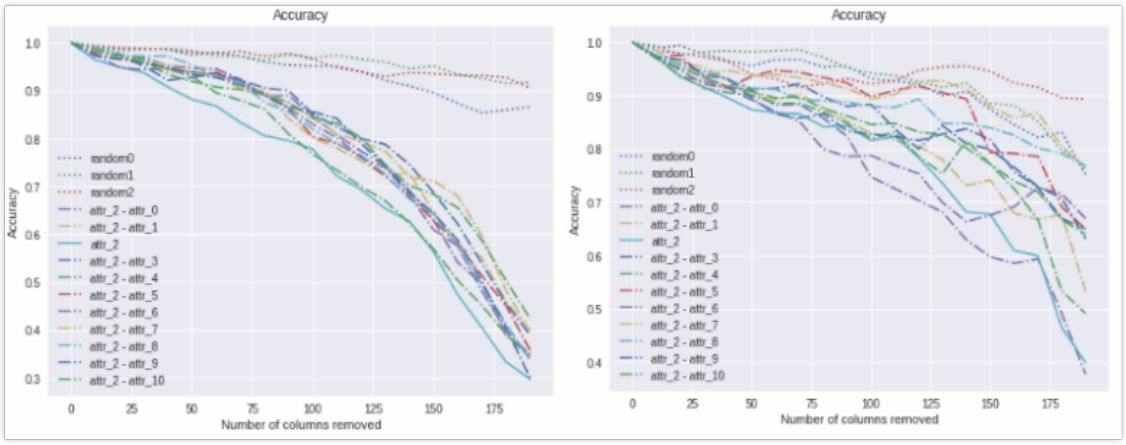


Figure 6.39: The accuracy when removing embedding columns for "credit card" documents based on attribution scores by LRP (left) and saliency map (right)

Similar to the previous experiment, the attribution scores and attribution differences have a more significant impact on the model accuracy. From the experiment in Figure 6.34 with LRP attribution scores, the attribution scores for the true class ("attr_2") has caused lower accuracy. However, the same observation cannot be made from the experiment with attribution scores generated by saliency map. From the accuracy axis, the experiments with LRP attribution scores have also achieved lower accuracy for attribution scores and attribution differences.

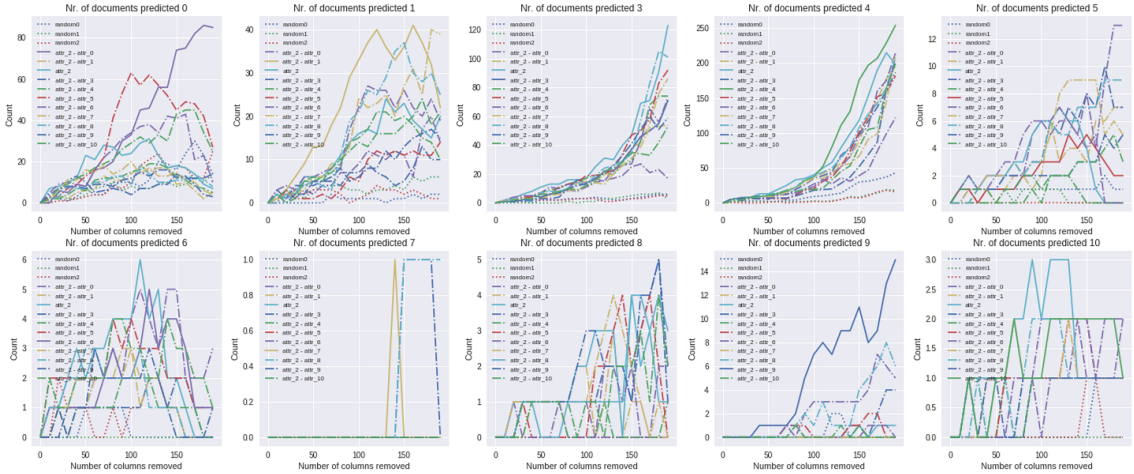


Figure 6.40: The mis-classification count when removing embedding columns for "credit card" documents based on attribution scores by LRP

The numbers of documents classified as "5" - "10" are too small. Therefore, the results from these misclassifications are not convincing. From "Nr. documents predicted 0", "attr_2 - attr_0" seems to have resulted in more document predicted as "0". The same trend, though not as obvious as the previous experiments in Figure 6.32, can be observed in class "1", "3", "4" as well.

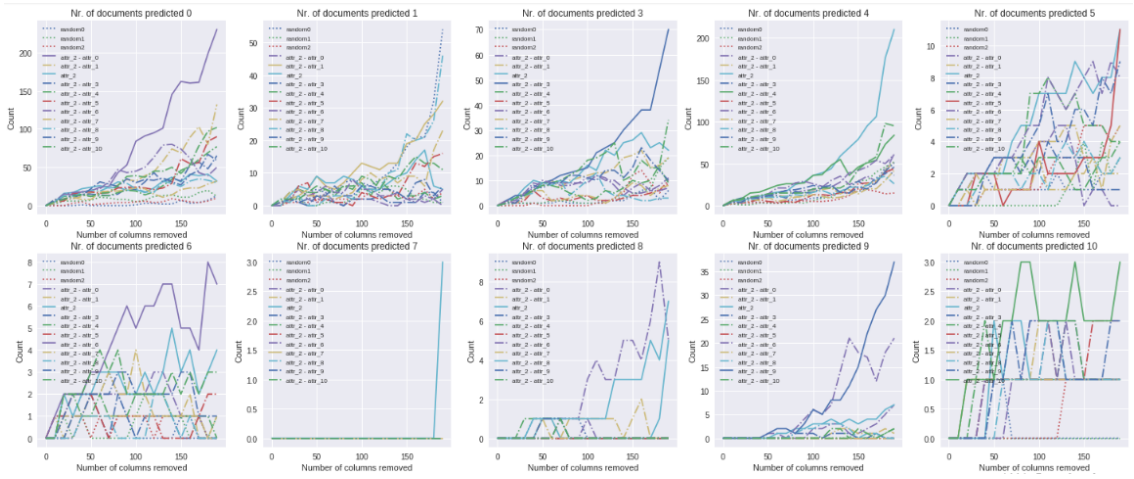


Figure 6.41: The mis-classification count when removing embedding columns for "credit card" documents based on attribution scores by saliency map

When using the attribution scores generated by saliency map, the numbers of documents classified as "5" - "10" are also too small. Thus, they are not considered here either. From "Nr. documents predicted 0", "attr_2 - attr_0" seems to have resulted in more document predicted as "0". The

same trend can be observed in class “3“, “4“ as well but not in class “1“.

Result analysis

- The experiments can show the differences in saliency map and LRP

  From measuring the changes in accuracy when removing embedding columns, the approaches based on LRP attribution scores/differences have caused a more severe damage than saliency map. The reason is that based on LRP, the columns with positive contributions are removed. While based on saliency map, the columns with large impacts, both positive and negative, are removed. Therefore, the changes in the predicted probability for the true label does not change as much.

- LRP is able to offer more local explanations

  By comparing the number of misclassifications for each class, the approaches based on LRP attribution differences are able to alter the predictions to be in favor of one specific class. However, saliency map attribution differences are not always able to achieve that.

6.4.3 Conclusions

With regard to research question 1, our evaluations were conducted without deleting any words. Each column in the embedded documents was considered a feature instead of a word.

To answer research question 4, a) the evaluations results were measurable and comparable. b) The evaluation was able to determine whether important features were assigned higher attribution values. For LRP, the accuracy decrease was obvious when most important positive/negative features were removed. For saliency map, it was not observed in binary classification tasks. However, on the US consumer finance complaint dataset, the decrease in accuracy was also apparent when features with high attribution values for their true class were removed. c) The results reflect the difference of saliency map and LRP on whether the attribution scores are signed.

6.5 Feature Removing Experiments with Convolutional Filters

6.5.1 Experiments on Yelp Review Dataset

Experiments on Task “stars“

1. Attribution scores on the “Concatenate layer“

   The attribution scores for a document on the “Concatenate layer“ is a numeric vector. To investigate whether the scores on a node are similar for different documents from a certain class, the attribution scores are visualized.

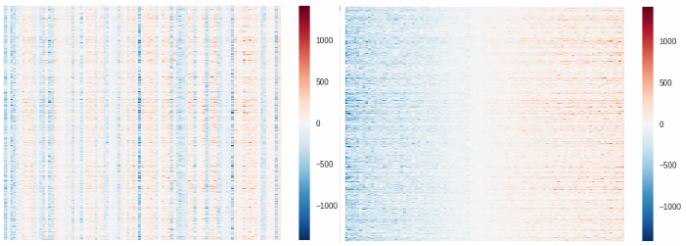


Figure 6.42: Heatmap of attribution score (LRP) sum for “0“ stars on n-gram feature representations

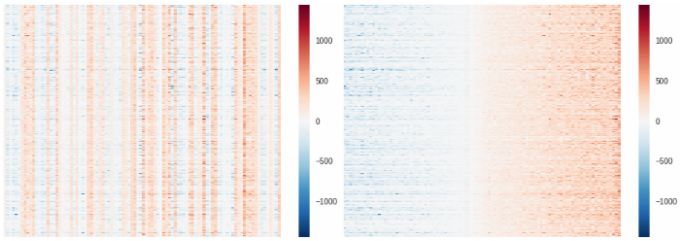


Figure 6.43: Heatmap of attribution score (LRP) sum for “1“ stars on n-gram feature representations

The above figures are heatmaps of attribution scores on the “Concatenate layer“. Each row represents the scores for a document. Each column is the attribution scores for a node in “Concatenate layer“. Documents from different classes are visualized separately. The attribution scores ordered by the sum of each column are also plotted.

From the heatmaps, fro a particular node, the attribution scores are very similar. From the ordered heat map, a clear separation between positive and negative attributions can be observed.

The same trend is also shown in the attribution scores generated by saliency map.

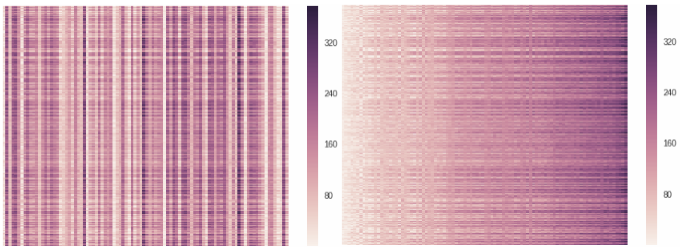


Figure 6.44: Heatmap of attribution score (saliency map) sum for “0“ stars on n-gram feature representations

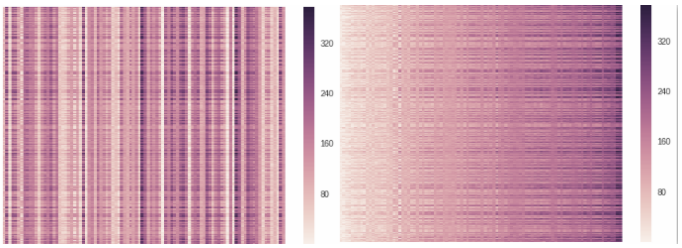


Figure 6.45: Heatmap of attribution score (saliency map) sum for "1" stars on n-gram feature representations

2. Evaluate the attribution scores by removing filters

   Three different experiments are conducted on data from different classes:

   - Remove filters with the smallest attribution scores. This experiment is conducted on documents classified as "0". The purpose is to remove filters with negative contributions so that the negative documents will be misclassified as positive. When using saliency map attribution scores, the filters with the largest attribution scores are removed since the attributions are not signed.

   - Remove filters with the largest attribution scores. This experiment is conducted on documents classified as "1". The purpose is to remove filters with positive contributions so that the positive documents will be misclassified as negative.

   - Remove filters with the smallest absolute attribution scores. This experiment is conducted on documents classified as "0" and "1". The purpose is to remove filters with little contributions so that the accuracy stays approximately the same level.

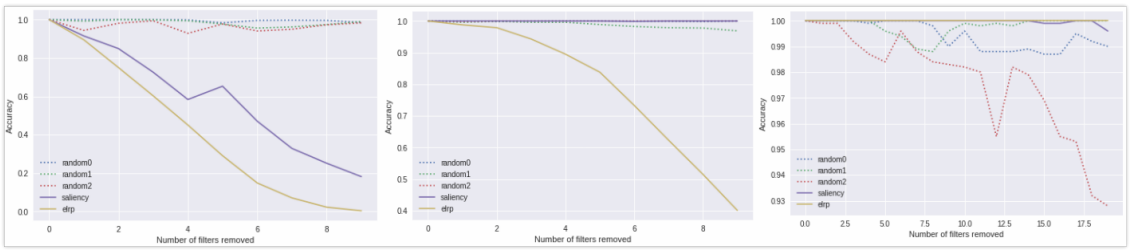


Figure 6.46: Model accuracy when removing filters with the smallest(left), largest(mid) and smallest absolute attributions(right) on task "stars"

From the results from using LRP attribution scores, we can see that after removing 6 filters with the smallest attribution scores, the accuracy on documents labelled "0" became lower than 20% (around 50% for saliency map). After removing 6 filters with largest attribution scores, the accuracy on documents labelled "1" decreased to around 75%.

Experiments on Task "funny"

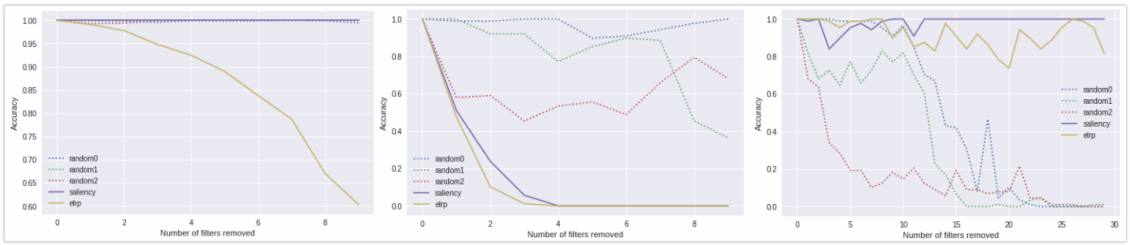


Figure 6.47: Model accuracy when removing filters with the smallest(left), largest(mid) and smallest absolute attributions(right) on task "funny"

From the results from using LRP attribution scores, after removing 6 filters with the smallest attribution scores, the accuracy on documents labelled "0" became lower than 85%. After removing 6 filters with largest attribution scores, the accuracy on documents labelled "1" decreased to 0% (same for saliency map). When removing 1-4 filters with largest attribution scores, the accuracy decreased faster when using LRP.

Result Analysis

- For both task "stars" and "funny", removing filters based on both LRP and saliency map attributions have successfully preserved the model accuracy better than random approaches.

- Compared to random approaches, the model accuracy drops more when removing filters with the largest saliency map attributions on documents classified as "0" in "stars" and "1" in "funny". However, for documents classified as "1" in "stars" and "0" in "funny", the accuracy change is not significantly different from random approaches. One of the reasons is that attributions generated by saliency map are not signed. In task "stars", the filters with large attributions are mostly negative to the prediction. In task "funny", the filters with large attributions are mostly positive to the prediction.

- Compare to other approaches, removing filters based on LRP attribution scores was able to lower the accuracy more and faster.

6.5.2 Experiments on US Consumer Finance Complaint Dataset

In this experiment, the differences in attribution scores between the true class and other classes are computed as well. The dataset used for the following experiment is the same as 6.3.2.

Experiments on Documents Labelled "bank account or service"

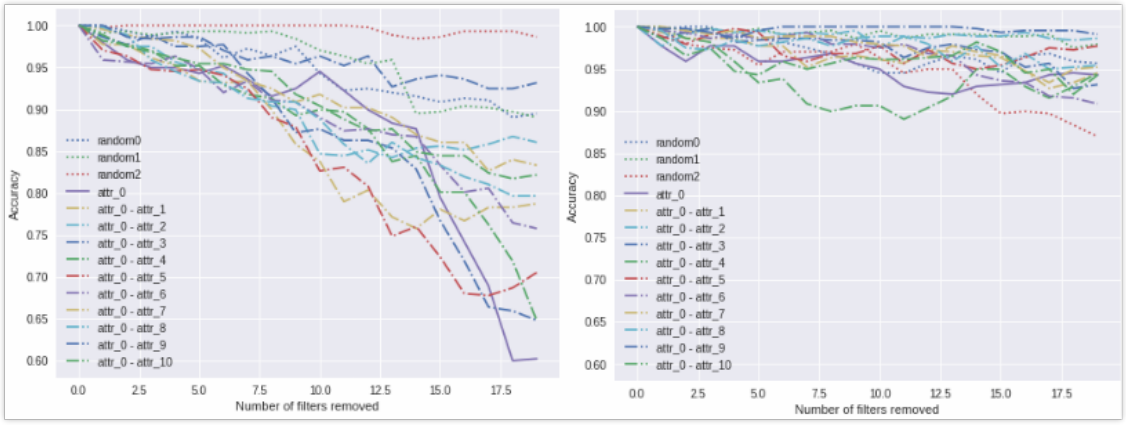


Figure 6.48: Model accuracy when removing filters for "back account or service" documents based on attribution score by LRP (left) and saliency map (right)

Above is the model accuracy changes when removing the filters with the largest attribution scores for the true class or attribution difference scores. From the left figure, the attributions are generated by LRP. Compared to randomly removing the embedding columns, the approaches based on attribution scores or attribution differences have all resulted in a larger drop in accuracy. From the right figure, which is on the same axis with the left figure, the attributions are generated by saliency map. In comparison, the accuracy changes, though larger than random approaches, are smaller than accuracy changes when using LRP attribution scores.

When removing filters with large attribution scores, the model accuracy will drop, which means that some documents are misclassified. Therefore, the following experiments are conducted. Every time a filter is removed, the number of documents misclassified for each class are recorded.

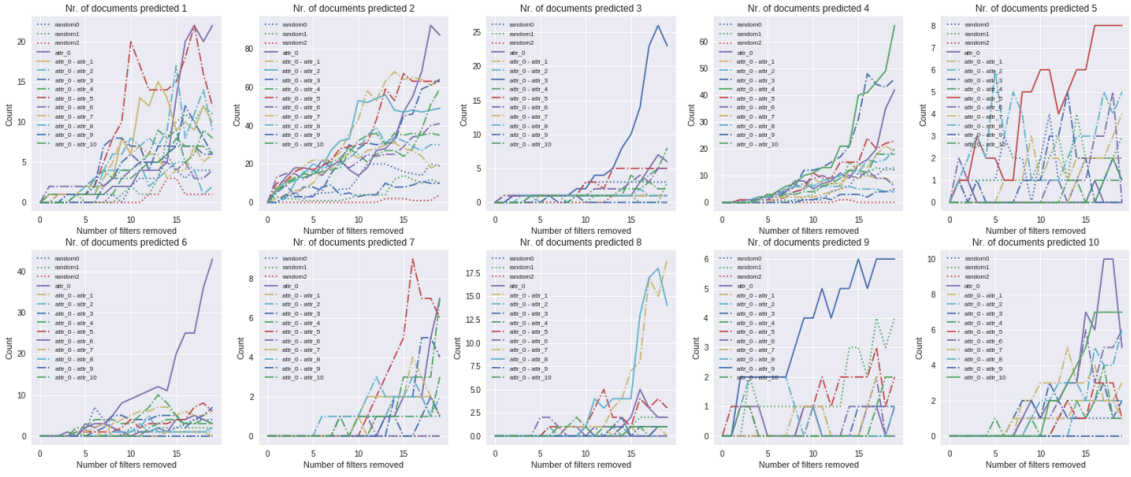


Figure 6.49: The mis-classification count when removing embedding columns for "bank account or service" documents based on attribution scores by LRP

In the above figure, in "Nr. of documents predicted 3", "attr_0-attr_3" (differences in attribution scores between class 0 and class 3) has a much higher number of documents misclassified as "3" than other approaches. The same trend can be observed in most other classes.

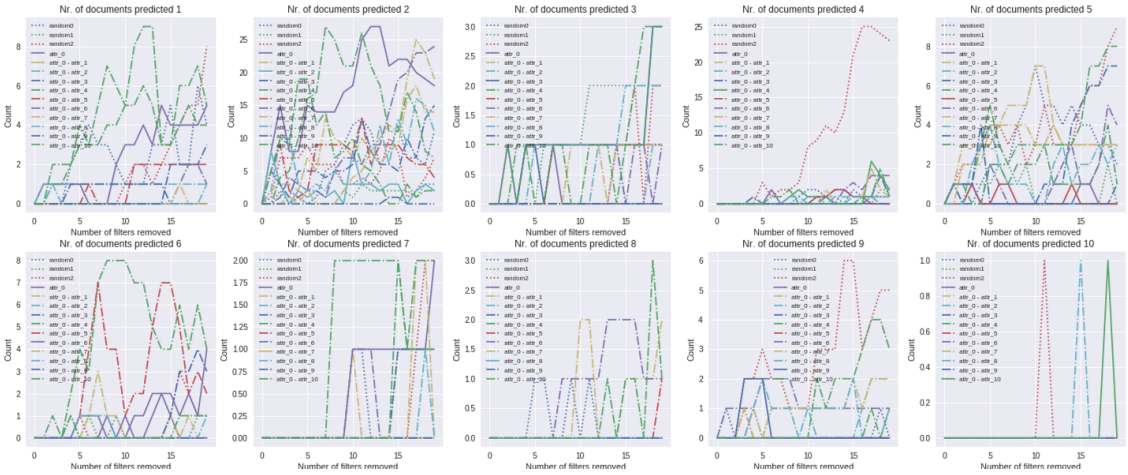


Figure 6.50: The mis-classification count when removing embedding columns for "bank account or service" documents based on attribution scores by saliency map

When using attributions generated by saliency map, the results are different. In "Nr. document predicted 2","attr_0-attr_2" has sometimes resulted in a higher number of documents classified as "2". However, such a trend was not shown in all other classes. What was observed when using LRP attributions is not observed here.

Experiments on Documents Llabelled "credit card"

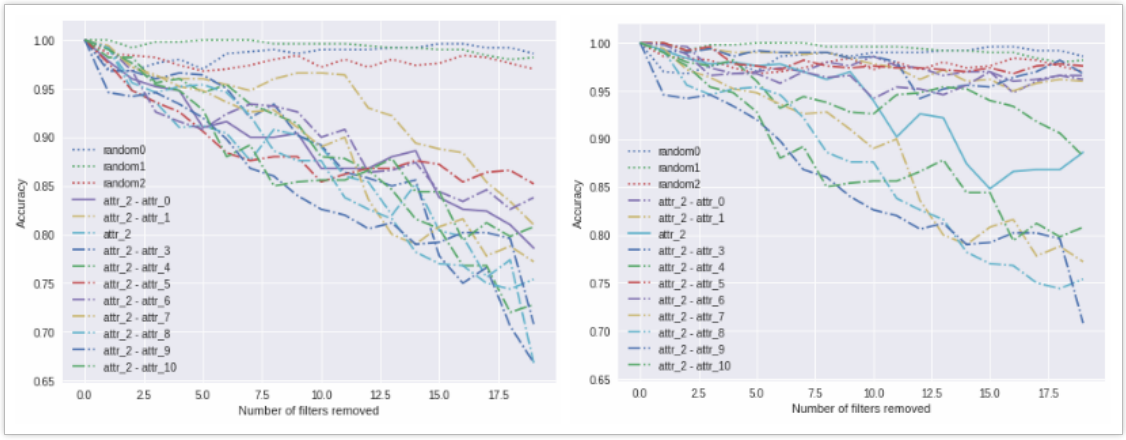


Figure 6.51: Model accuracy when removing filters for "credit card" documents based on attribution score by LRP (left) and saliency map (right)

Above is the model accuracy changes when removing the filters with the largest attribution scores for the true class or attribution difference scores. From the left figure using LRP, it is easy to observe that the approaches based on attribution scores or attribution differences have all resulted in a more significant drop in accuracy. From the right figure using saliency map, which is on the same axis with the left figure. In comparison, the accuracy changes, though larger than random approaches, are overall slightly smaller than accuracy changes when using LRP attribution scores.

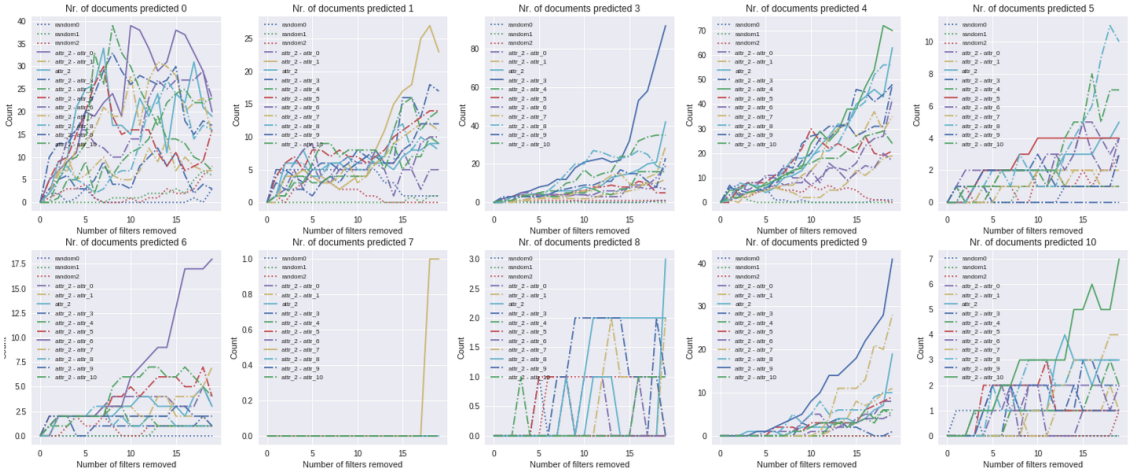


Figure 6.52: The mis-classification count when removing embedding columns for "credit card" documents based on attribution scores by LRP

In the above figure, in "Nr. of documents predicted 3", "attr_2-attr_3" (differences in attribution scores between class 2 and class 3) has a much higher number of documents misclassified as "3" than other approaches. The same trend can be observed in most other classes.

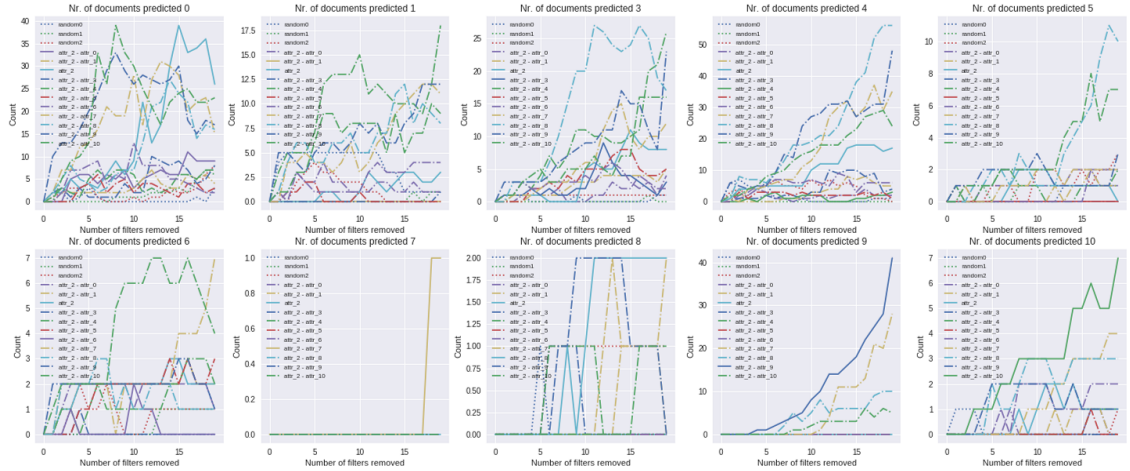


Figure 6.53: The mis-classification count when removing embedding columns for "credit card" documents based on attribution scores by saliency map

When attributions are generated by saliency map, the results are different. In "Nr. document predicted 9","attr_2-attr_9" has sometimes resulted in a higher number of documents classified as "9". But such trend was only shown in class "9" and "10". What was observed when using LRP attributions is not observed here.

Result Analysis

- The experiments are able to show the differences in saliency map and LRP From measuring the changes in accuracy when removing filters, the approaches based on LRP attribution scores have caused a more severe decrease in accuracy. One of the reason is that based on LRP, filters with positive contributions were removed. While based on saliency map, filters

with both positive and negative impacts were removed, which offers some balances in the accuracy change.

- LRP is able to offer more local explanations From comparing the misclassification counts for each class, removing filters based on LRP attribution differences were able to alter the predictions to be in favor of one particular class. However, saliency map attribution difference has failed to achieve the same.

6.5.3 Conclusions

For research question 2, the evaluations in this section did not make any changes to the input data. Instead, changes were made to the neural network. By removing the convolutional filters, evaluations were conducted measuring the accuracy changes.

To answer research question 4, a) the evaluations results were measurable and comparable. b) The evaluation was able to determine whether important features were assigned higher attribution values. For LRP, the accuracy decrease was obvious when most important positive/negative features were removed. For saliency map, it was not observed in binary classification tasks. However, on the US consumer finance complaint dataset, the decrease in accuracy was also obvious when features with high attribution values for their true class were removed. c) The results reflect the difference of saliency map and LRP on whether the attribution scores are signed.

6.6 Visualizations

In this section, examples will be used to demonstrate how the visualization tool can be useful for analyzing the model's predictions.

An example misclassification analysis: a document with true class "Consumer loan" was misclassified as "Bank account or service".

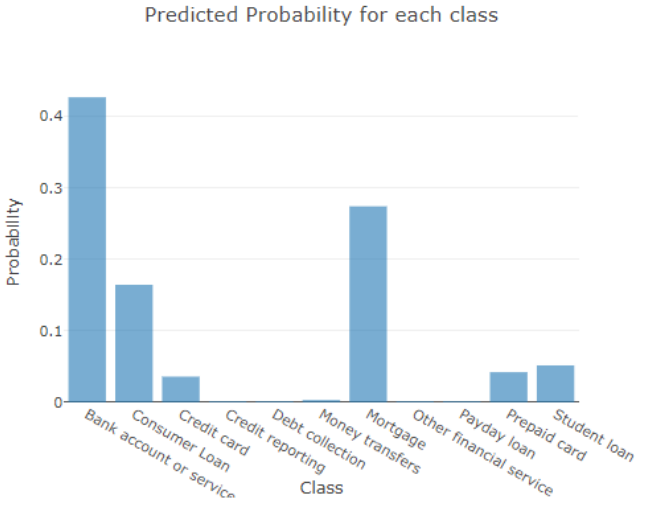


Figure 6.54: The predicted probability of the mis-classified document

From the predicted probabilities for each class, probabilities for class "bank account or service" and "mortgage" are higher than the probability for true class "consumer loan". Therefore, attributions

from these three classes will be analyzed.

First of all, the highlights of the words will be inspected.

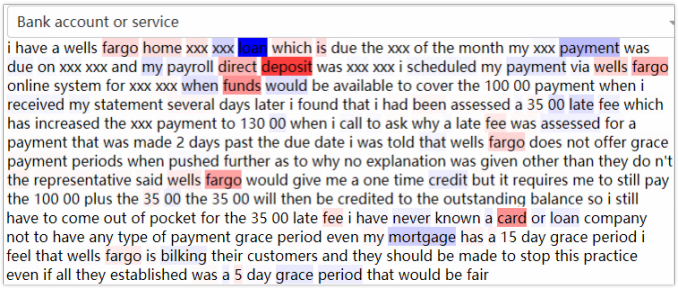


Figure 6.55: The highlighted text for class “bank account or service“

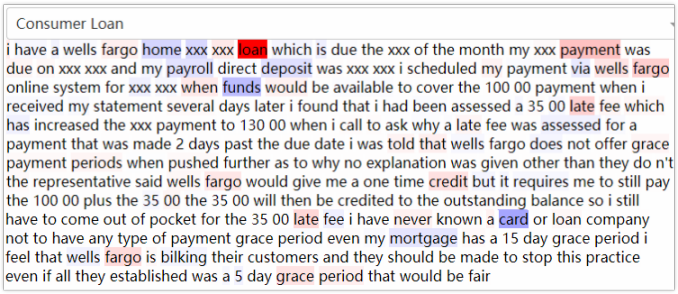


Figure 6.56: The highlighted text for class “consumer loan“

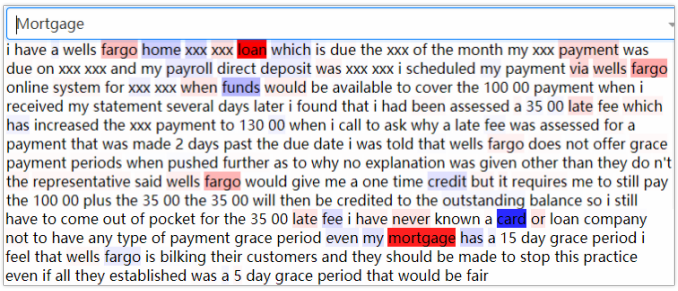
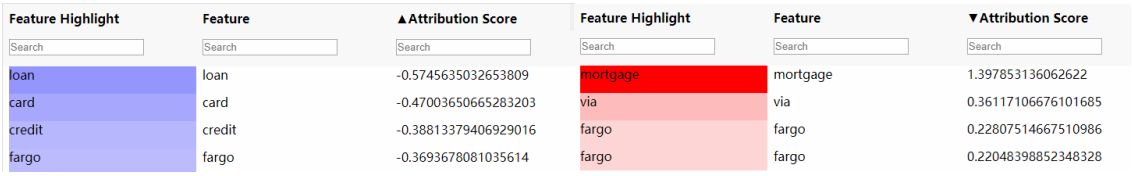


Figure 6.57: The highlighted text for class “mortgage“

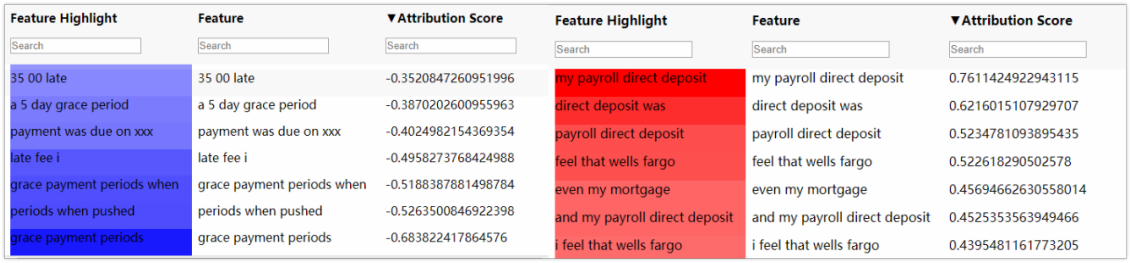
From the highlights on text, it is easy to tell that the words positive for class “bank account or services“ are “deposit“, “card“ and “funds“ while the negative ones are “loan“, “payment“ and “mortgage“. The words positive for class “mortgage“ are “loan“ and “mortgage“ while the negative ones are “cards“, “funds“ and “home“. For the true class “consumer loan“, positive words are “loan“ and “payment“, negative words are “cards“ and “funds“.



Feature Highlight	Feature	▲Attribution Score	Feature Highlight	Feature	▼Attribution Score
Search	Search	Search	Search	Search	Search
loan	loan	-0.5745635032653809	mortgage	mortgage	1.397853136062622
card	card	-0.4700365066528320 3	via	via	0.36117106676101685
credit	credit	-0.38813379406929016	fargo	fargo	0.22807514667510986
fargo	fargo	-0.3693678081035614	fargo	fargo	0.22048398852348328

Figure 6.58: Attribution differences between class “mortgage“ and the true class “consumer loan“

To investigate further, the attribution differences between words will be examined as well. It can be observed that even though “card“, “loan“ are negative to both class “consumer loan“ and “mortgage“. Compared to “mortgage“, it contributes more to the true class “consumer loan“. The word “mortgage“ is mainly the reason why it was predicted to be more likely “mortgage“ over “consumer loan“.



Feature Highlight	Feature	▼Attribution Score	Feature Highlight	Feature	▼Attribution Score
Search	Search	Search	Search	Search	Search
35 00 late	35 00 late	-0.3520847260951996	my payroll direct deposit	my payroll direct deposit	0.7611424922943115
a 5 day grace period	a 5 day grace period	-0.3870202600955963	direct deposit was	direct deposit was	0.6216015107929707
payment was due on xxx	payment was due on xxx	-0.4024982154369354	payroll direct deposit	payroll direct deposit	0.5234781093895435
late fee i	late fee i	-0.4958273768424988	feel that wells fargo	feel that wells fargo	0.522618290502578
grace payment periods when	grace payment periods when	-0.5188387881498784	even my mortgage	even my mortgage	0.45694662630558014
periods when pushed	periods when pushed	-0.5263500846922398	and my payroll direct deposit	and my payroll direct deposit	0.4525353563949466
grace payment periods	grace payment periods	-0.683822417864576	i feel that wells fargo	i feel that wells fargo	0.4395481161773205

Figure 6.59: Attribution differences between class “bank account or service“ and the true class “consumer loan“

By looking into the attribution score differences between class “bank account or service“ and the true class “consumer loan“, features “grace payment periods“, “periods when pushed“, “grace payment periods when“ and “late fee i“ contributes more positively to the true class. However, their influences are not as strong as features “my payroll direct deposit“, “direct deposit was“ and “payroll direct deposit“ that contributes more positively to “bank account or service“.

With the help of the visualization tool, it was observed from this specific example that the model was able to capture features that distinguish each class. However, the contribution from the features towards the true class was weaker.

Chapter 7

Conclusions

In this thesis, an evaluation framework was established for attribution-based explanations for neural network outputs on text classification tasks. A prototype of an explanation application was developed to demonstrate the potential usefulness in the business world. In the following sections, the contributions are concluded in terms of academic and business values. Then, the limitations and future work are discussed.

7.1 Contributions

7.1.1 Academic Contributions

In this thesis, the research problem of attribution-based explanations was formulated, including how the attribution scores were computed. The differences in different approaches were analyzed theoretically. Then, the limitations of the current evaluation approach on attributions were discussed, based on which research questions were asked. Possible solutions were also provided.

Before developing the evaluation, a clear definition of what qualified for a good evaluation for an attribution-based explanation was given. Then, based on the discussion on the research problems, a new evaluation framework on the attributions was established. Approaches were proposed to evaluate the attribution scores from word level, embedded document level and n-gram level, in accordance to the model's input layer, embedding layer and convolutional layer.

In the experiment, first, document representations were generated to evaluate whether the truly representative features for a document's true class were assigned a higher attribution score. The results, both qualitatively and quantitatively, were able to validate that when using attribution scores, information on the true class was amplified in the document representations. The evaluations were able to determine whether truly important feature was found with measurable results.

Then, feature removing experiments were conducted on both the embedding layer and convolutional layer to assess the influence of a feature (embedding column and convolutional filter) on the model output. Through the experiments, differences between saliency map and LRP were shown. The attribution scores generated by saliency map was not signed. Thus, it could not distinguish between positive and negative influence. When removing a positive feature in a binary classification, the predictions were not guaranteed to drift to be negative. When removing a feature more positive to a certain outcome than another outcome in a multi-class classification task, the predictions were not guaranteed to be more in favor of that certain outcome. The feature removing approaches with both embedded documents and convolutional filters were able to

determine whether truly important feature were found with measurable results. Also, they were able to reflect the "signed" difference between saliency map and LRP.

	Document representation	Feature removing with embedded documents	Feature removing with convolutional filters
Measurable	Yes	Yes	Yes
Whether truly important features are found	Yes	Yes	Yes
Whether it is able to show "signed" difference	No	Yes	Yes
Whether it is able to show "use input in backpropagation" difference	No	No	No

Table 7.1: Summarization for each evaluation approach

To summarize, several evaluation approaches were proposed to evaluate attribution scores on words, embedding columns and n-gram features. The motivation to avoid deleting words is achieved. Based on our experiments, the approaches were proven to be effective. It was evaluated whether the truly important features were found. The characteristics of different attribution-based approaches can be observed from the results. Our evaluation framework met our standards for a good evaluation for attribution-based explanations.

7.1.2 Business Contributions

The following contributions were made to demonstrate the potential usage of attribution-based explanations in the business world:

- Attribution scores on n-gram features were generated. Instead of assessing the influence of a certain word, it is also useful to learn the mutual contribution of a few words. By visualizing the attributions on n-gram features instead of words, people can have an understanding of the predictions on a higher level.
- Attribution differences in words or n-grams between a document's true class and other classes were calculated and visualized on the corresponding feature. In this way, people can inspect the reason why a model is in favor of a class over the true class. It is especially useful in investigating misclassifications.
- An interactive visualization tool was built based on the attribution (difference) scores of words and n-gram features of a multi-class classification task. Potentially, it can be applied to build analysis tools for real use cases in banking compliance research.

7.2 Limitations and Future Work

The limitations and future work of this thesis are as follows:

- The experiments were conducted on two public datasets. The business contribution will be higher if we apply real-world data to validate and improve the usefulness of attribution-based approaches.

- Only TextCNN model is used. Part of our experiments was model-specific, such as generating attribution scores on n-gram features. In the future, experiments on different models should be provided to produce more general results.
- In the problem set, text classification task was chosen. However, in reality, there are issues on various data and tasks that require explanations. Experiments on other problem sets should be conducted as well to gain more general conclusions.
- In our feature removing experiments, an assumption was made that each time a feature is removed, the attribution score ranking for the other features stays the same. However, this statement is not always right.

Bibliography

[1] Marco Ancona, Enea Ceolini, Cengiz Oztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *6th International Conference on Learning Representations (ICLR 2018)*, 2018. 7

[2] Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. " what is relevant in a text document?": An interpretable machine learning approach. *PloS one*, 12(8):e0181142, 2017. 7

[3] Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. Explaining recurrent neural network predictions in sentiment analysis. *arXiv preprint arXiv:1706.07206*, 2017. 7

[4] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003. 5

[5] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012. 1

[6] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016. 1

[7] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014. 5, 6

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[9] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*, 2015. 7

[10] Zachary C Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016. 6

[11] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 2017. 1

[12] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901. 14

[13] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 5

[14] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*, 2017. 10, 11

[15] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *arXiv preprint arXiv:1704.02685*, 2017. 6

[16] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 7, 10

[17] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014. 6

Appendix A

Appendix

A.1 Visualization when removing embedding columns on text using saliency map on task "stars"

When removing embedding columns based on the saliency map, the corresponding attribution scores are also removed. We have plotted the attribution scores on the original text.

Figure A.1: Attribution scores on text as removing embedding columns with the largest saliency attribution scores

Figure A.2: Attribution scores on text as removing embedding columns with the smallest saliency attribution scores

A.2 Classification results on document representations in task "stars"

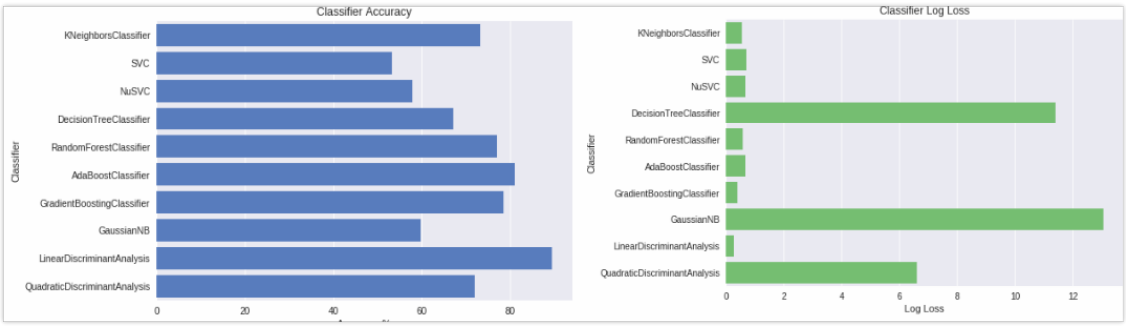


Figure A.3: Classification accuracy and loss on unweighted document representations for task "stars"

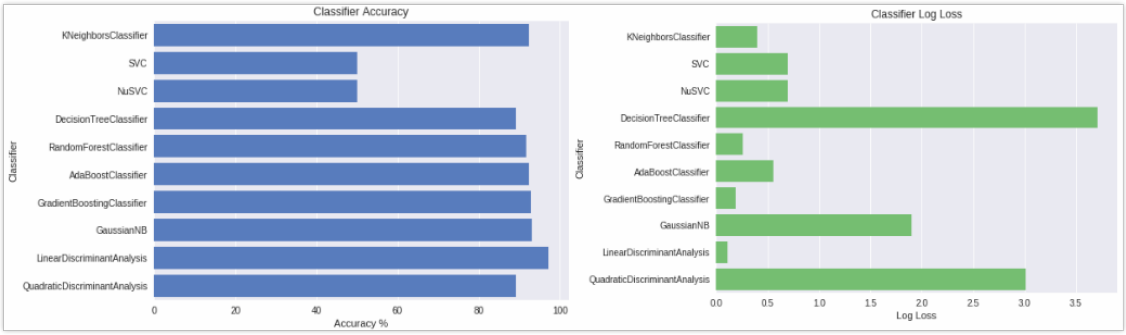


Figure A.4: Classification accuracy and loss on weighted document representations with LRP for task "stars"

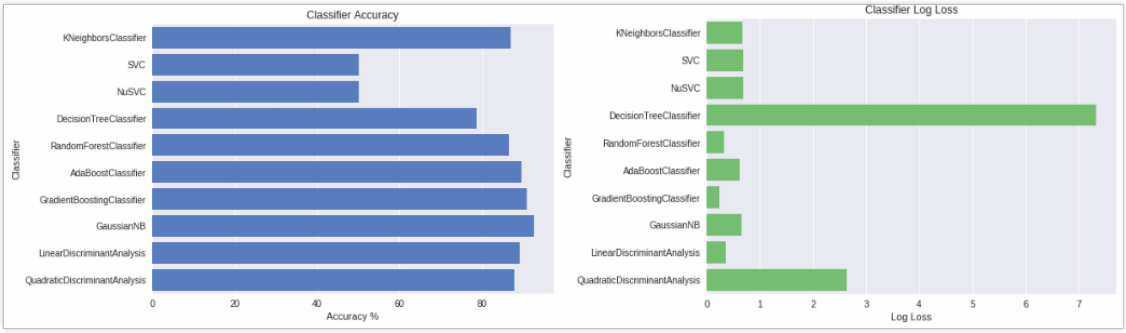


Figure A.5: Classification accuracy and loss on individually weighted document representations with LRP for task "stars"

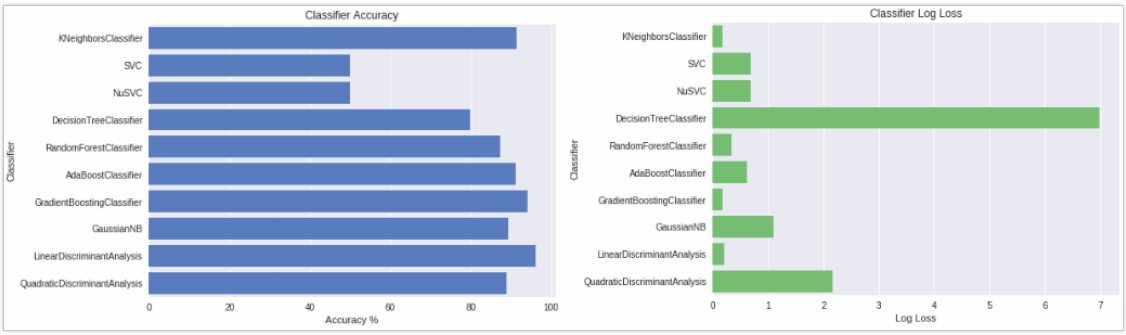


Figure A.6: Classification accuracy and loss on weighted document representations with saliency map for task "stars"

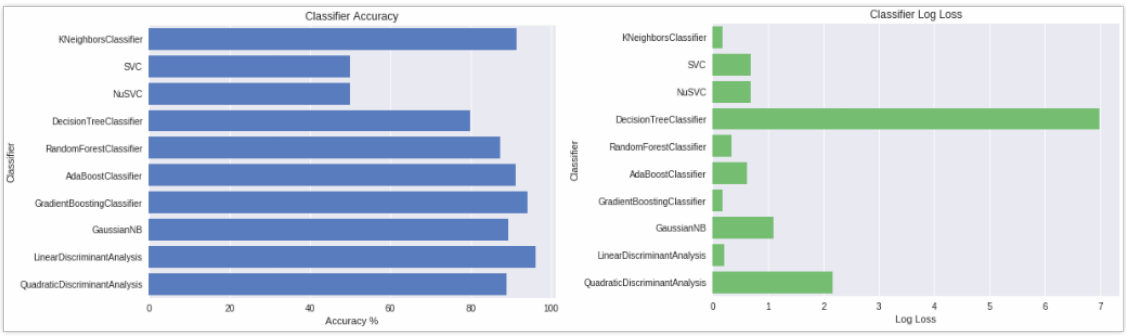


Figure A.7: Classification accuracy and loss on individually weighted document representations with saliency map for task "stars"

A.3 Classification results on document representations in task "funny"

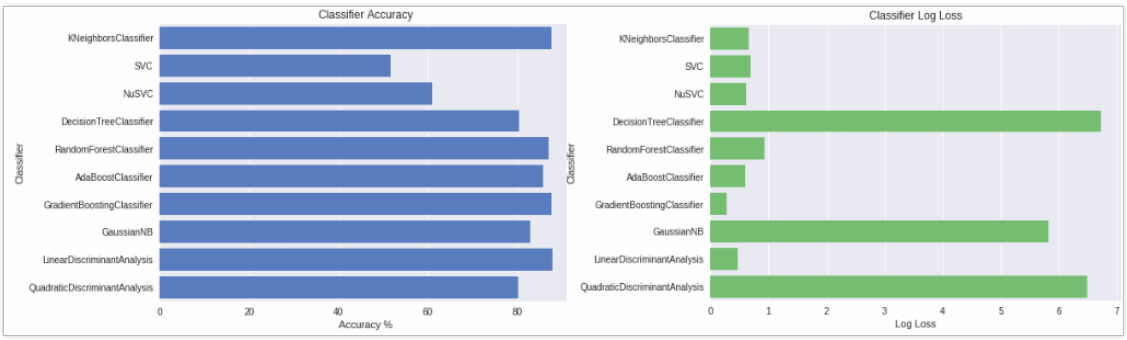


Figure A.8: Classification accuracy and loss on unweighted document representations for task "funny"

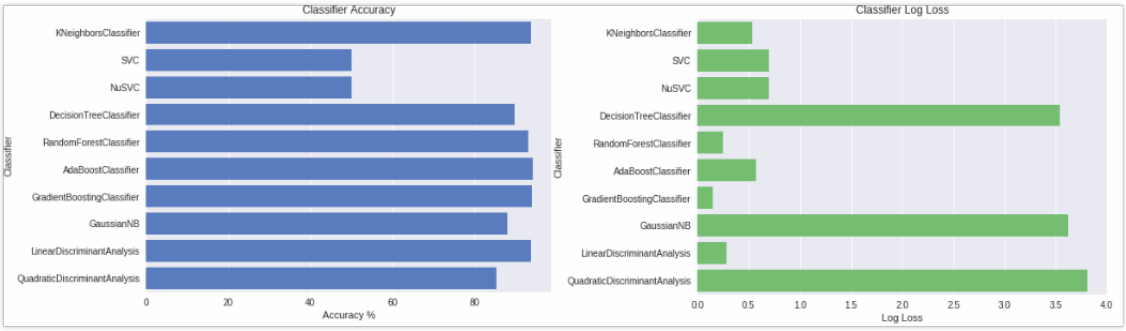


Figure A.9: Classification accuracy and loss on weighted document representations with LRP for task "funny"

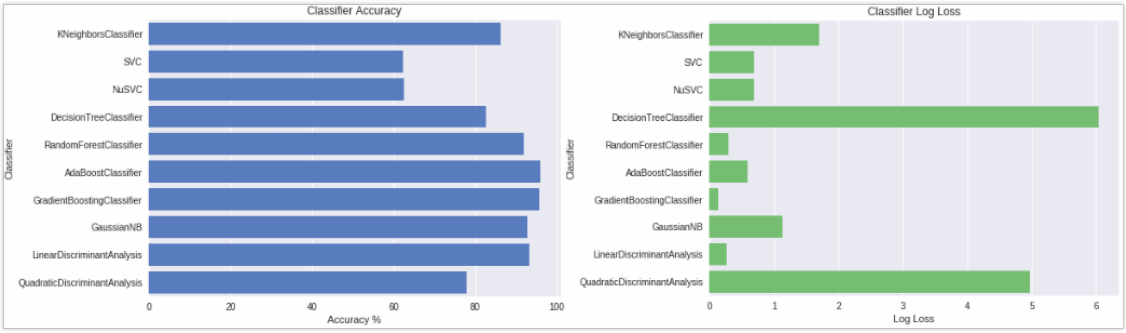


Figure A.10: Classification accuracy and loss on individually weighted document representations with LRP for task "funny"

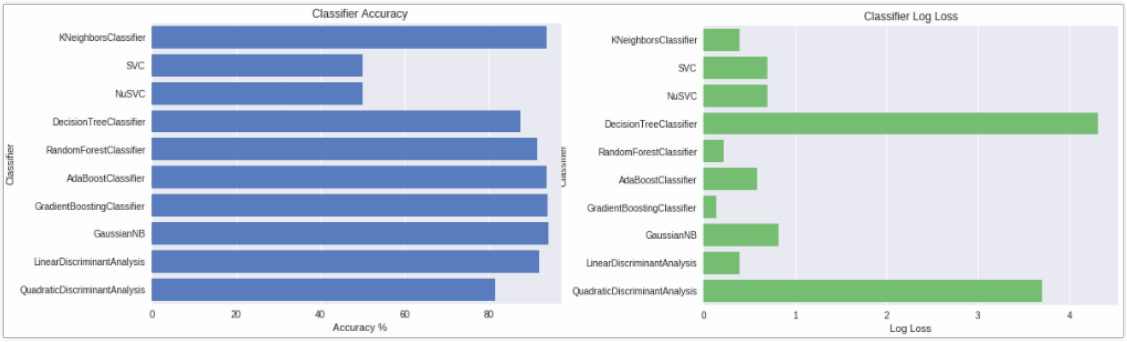


Figure A.11: Classification accuracy and loss on weighted document representations with saliency map for task "funny"

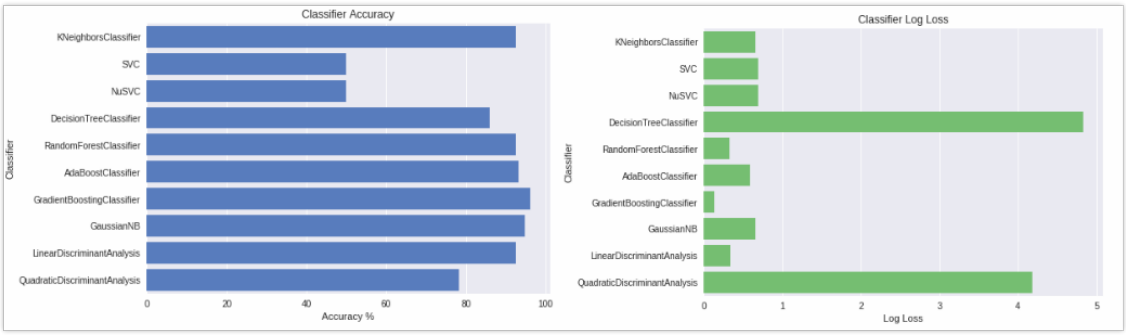


Figure A.12: Classification accuracy and loss on individually weighted document representations with saliency map for task "funny"

A.4 Classification results on document representations on US consumer finance complaint dataset

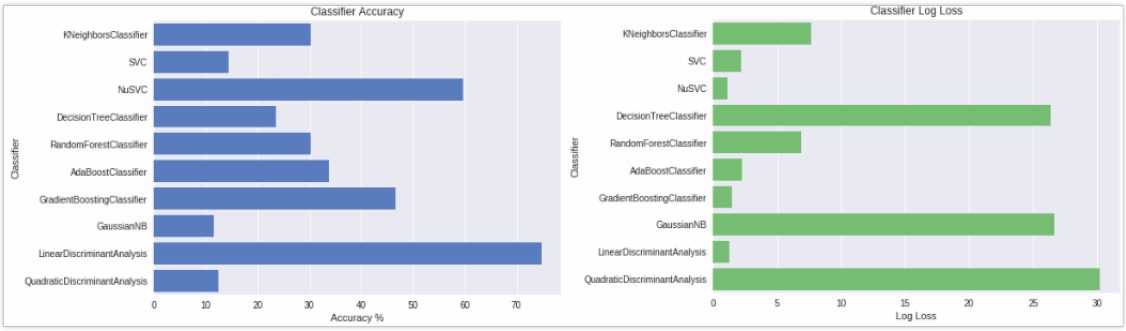


Figure A.13: Classification accuracy and loss on unweighted document representations

For unweighted documents, the accuracy is mostly below 50%. For some models, the accuracy is barely 20%.

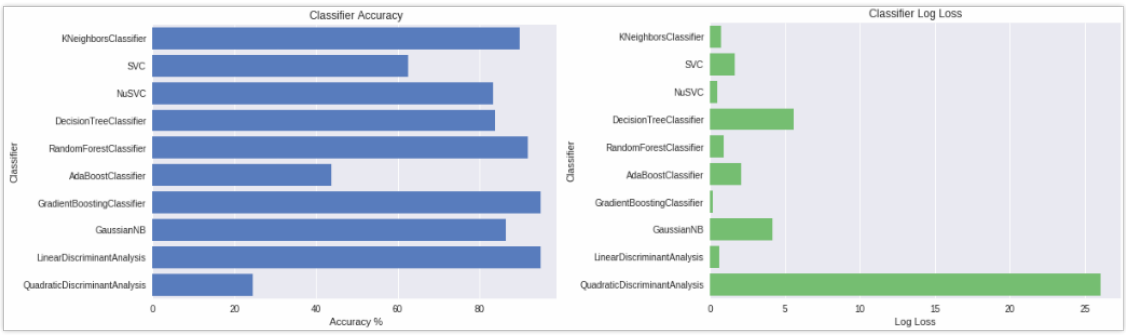


Figure A.14: Classification accuracy and loss on weighted document representations by LRP

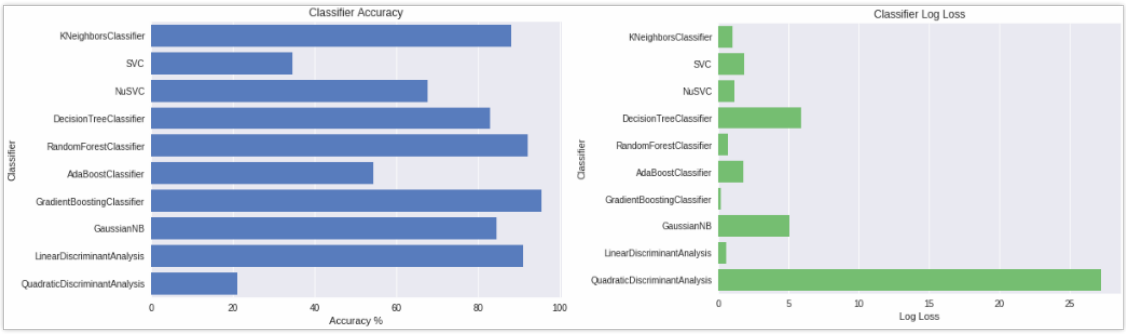


Figure A.15: Classification accuracy and loss on weighted individually representations by LRP

For weighted document representations both on word and on individual embedded feature by LRP, an obvious increase in accuracy and decrease for log loss is obvious for most models. For these two weighted representations, the figures showing accuracy and loss are on the same scale. Their results do not show substantial differences.

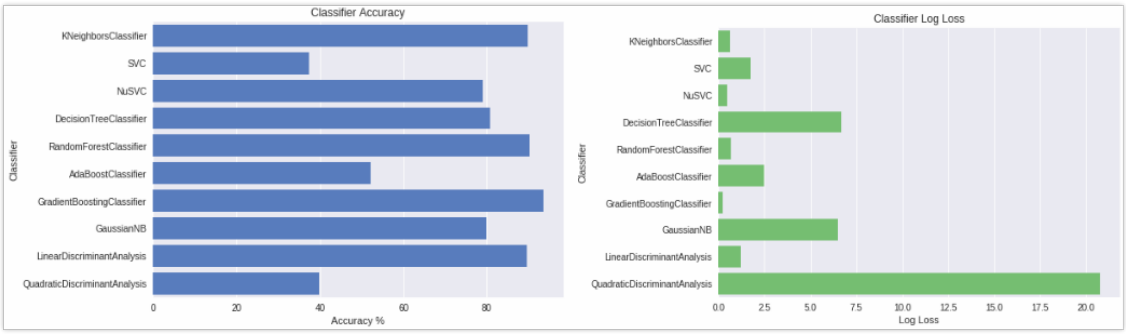


Figure A.16: Classification accuracy and loss on weighted document representations by saliency map

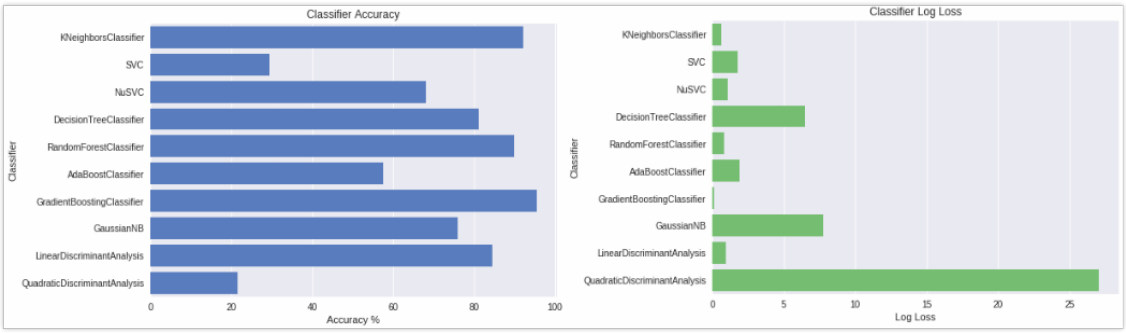


Figure A.17: Classification accuracy and loss on weighted individually representations by saliency map

When weighted by saliency map, the obvious increase in model performances is observed as well.