Bachelor scriptie

Text mining pathology reports

Names	:	Josien Visschedijk (s1010384)
Course	:	Bachelor scriptie
Course code	:	NWI-IBC
Supervisor	:	D. Hiemstra
Second supervisor	:	T. Laarhoven
Course coordinator	:	P. Achten
Date	:	January 21, 2020

Abstract

Aim This thesis describes the classification of pathology reports using text mining algorithms. The aim of the thesis is to analyse the pathology reports and obtain the highest possible performance - measured by an F1 score - in classifying these reports for 32 glomerular diseases. This is done in order to link a diagnosis (or diagnoses) to a pathology report and support nephrologist in their decision making process.

Method With two classifiers and an alteration of several parameters, 63 separate classifiers are designed. Of these models, 49 models used a decision tree classifier, and 24 models used a neural network classifier. Examples of the alteration of parameters is the use of binary- or multilabel classification, or the use of different feature selection methods.

Results The mean F1 score of the top five models using a decision tree classifier is ±0.3. The mean F1 score for the neural network classifier is ±0.4. There is a relation between the occurrence of a specific glomerular disease and the F1 score. Predominant diseases have an F1 score of ±0.8, whereas rare diseases have an F1 score of ±0.1. The best mean F1 scores are achieved with a 3-layered neural network, using multilabel classification and a tensor flow implementation.

Conclusion With an overall mean F1 score of ±0.4, the classifiers are not sufficient in fully supporting nephrologists in their decision making process when classifying pathology reports. However, for predominant glomerular diseases, the classifiers could serve as a supporting factor. It is recommended to conduct further research into setting up a categorization system with pre-defined values for better results in classifying pathology reports.

Acknowledgements

This thesis is written as part of the bachelor Computing science at the Radboud University, in Nijmegen. This thesis describes designing text mining algorithm(s) to classify pathology reports. These pathology reports come from the nephrology department of the Radboud UMC, in Nijmegen.

During the process of writing the thesis and designing the text mining algorithms, I learned a lot about not only the used algorithms, but also about the potential of these algorithms. I think in general, designing a text mining algorithm for (semi-)unstructured data is a huge challenge. However, with all the possible (artificial) intelligence and classification algorithms, I am excited to see what the future holds for combining machine learning with medical data.

I would like to thank my supervisor from the Radboud University, Djoerd Hiemstra for his valuable insights. Wynand Alkema, bio-informatician in the Radboud UMC, also had great ideas in designing text mining algorithms, for which I am grateful. Finally, I would like to thank Jan van den Brand and Jack Wetzels for a lot of background information about glomerular diseases and the processes within the nephrology department.

Table of contents

1 Introduction

Every day, hundreds of people visit the hospital, hoping to get their medical problems resolved. After their visit, their medical records are updated with new information. The nephrology department within the Radboud University Medical Centre (UMC) focuses on glomerular diseases and renal transplantation [77]. In case of suspicious tissue, a biopsy is taken for further research. After a nephrologist analyses the biopsy, a pathology report is constructed. This report contains the findings and a final diagnosis or diagnoses. In the past, all these pathology reports were printed and archived into files. In recent years, an effort was made to digitalize these files. This raised the question: is there an efficient way to analyse these pathology reports by means of a text mining algorithm?

Analyzing the pathology reports can offer benefits for the nephrology department of the Radboud UMC. First off, a clear diagnosis can be linked to a pathology report, which is at the moment not the case. This can bring more insight in the distribution of the glomerular diseases and similarities between groups of patients. Secondly, based on the findings in the pathology reports, a classification algorithm can be designed in order to predict the right diagnosis or diagnoses. This can support nephrologists in their decision making process in stating a diagnosis or diagnoses. Finally, certain important features can be extracted, to make pre-defined categories on which a pathology report is assessed. This can in the future be of great help in image processing. With the right features being extracted, an image classification algorithm can be designed, such that a histological examination is done without the intervention of a nephrologist.

In this thesis, the focus will be on linking a clear diagnosis (or diagnoses) to a pathology report, designing a classification algorithm, and making pre-defined categories.

As of right now, all files have been scanned and saved as one pdf file. A text mining algorithm was designed by a bioinformatician to extract certain features of nephrology diseases. This led to an improvement, but the goal to analyse these reports further still remained.

The pathology reports can be divided in an *analysis* section, and a *conclusion* section. The analysis section describes the findings of the histological examination. The conclusion section describes one or more stated diagnoses of the pathology report. In the conclusion section, the diagnosis or diagnoses are described, rather than clearly stated. This brings a challenge to design a text mining algorithm.

Within the nephrology department, there are three ways of analyzing a biopsy:

- Light microscopy;
- Electron microscopy;
- Immunofluorescence;

Not all these methods are used for every biopsy and thus not included in every analysis section of a pathology report. This makes that the pathology reports do not always have the same structure, which brings yet another challenge in classifying pathology reports.

In this bachelor thesis, the main goal is to design text mining algorithms which can analyse and classify these pathology reports. The text mining algorithms are based on two classifiers: decision tree classifier and neural network classifier.

The main question that will be answered in my bachelor thesis is:

> "Is it possible for a predictive algorithm to get to the same diagnosis as a human being when analyzing nephrology pathology reports?"

This main question will be answered based on the following sub questions:

1. What is the main structure of a pathology report and how is this of influence for a text mining algorithm?
2. What text mining algorithms are there to perform classification tasks on medical documents?
3. What is the diagnosis or are the diagnoses of each nephrology pathology report?
4. What F1 score can a text mining algorithm achieve in making a diagnosis, based on pathology reports?
5. What features are most important in stating a diagnosis?

The structure of this bachelor thesis is as follows. First off, the theoretical framework will be elaborated. The background of glomerular diseases, the current situation and different text mining algorithms used in the healthcare sector will be described. In the third section, the methodology of designing the classification models will be described. The fourth section describes the evaluation of the classifier models. The fifth section describes related work, where other text mining algorithms of clinical documents are elaborated. Finally, in the last section, the conclusions will be described and advise for follow up studies will be elaborated.

2 Theoretical Framework

In this theoretical framework, background information regarding glomerular diseases, the current situation in the Radboud UMC and text mining algorithms will be elaborated.

2.1 Glomerular diseases

The glomeruli (singular = glomerulus) are responsible for filtrating blood and the formation of urine. These relatively small organs are vital for maintaining a healthy body. Kidneys exist of a lot of tiny filters, called nephrons. Each of such a nephron exists of one glomerulus (see figure 1). The kidney itself exists of about 1 million nephrons. [28].

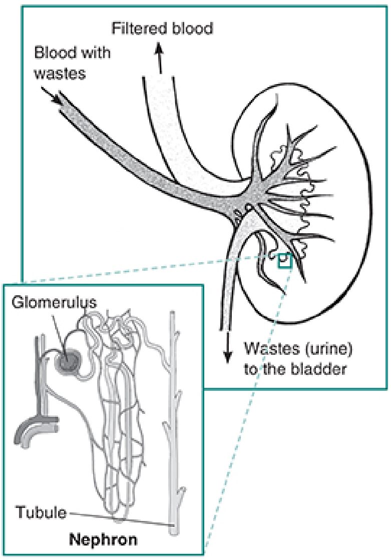


Figure 1: A single nephron with a glomerulus, from [59].

The glomeruli are responsible for the actual blood filtering. Water and waste fluids will be filtered out of the blood, which forms urine [28]. The glomerulus is encapsulated by Bowman's capsule, which exists of epithelial cells. The glomerular filter exists of the following three structures: capillary endothelium, a basement membrane and visceral epithelial cells (podocytes). The podocytes rest on the basement membrane. Within the glomerular capillary, there is tissue called mensangium. The cells of this tissue perform contraction and relaxation, which respectively leads to the decrease and increase of the filtration surface. [58]. In figure 2, the structure of the glomerular capillary is shown.

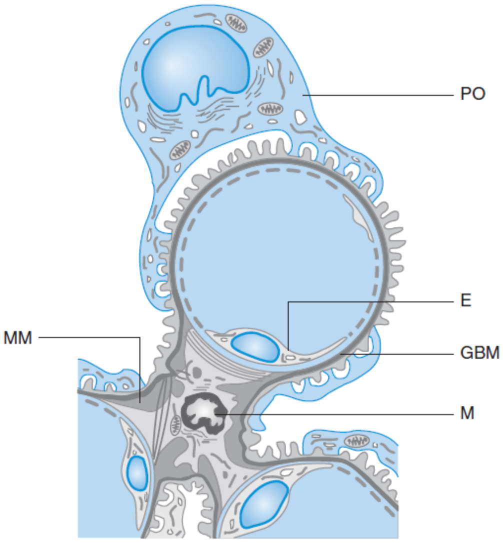


Figure 2: Structure glomerular capillary; PO = podocyt, MM = mesangial matrix, M = mesangial cell, GBM = glomerular basement membrane, E = endothelian cell [58].

Unfortunately, there are a lot of factors which can affect the kidneys and which are responsible for glomerular diseases. A glomerular disease is a disease of the kidneys, due to damage of the glomeruli [58].

Within the Netherlands, there are a lot of people suffering from glomerular diseases. Approximately 1.7 million people are suffering from chronic glomerular diseases. In addition, only 60% of those people actually are aware of these failures. Glomerular diseases are often noticed when just 30% of the kidneys are actually functioning, because then symptoms will occur [57]. When certain abnormalities are not detected in time, there is a chance that kidney failures occur.

There are certain factors that indicate a glomerular disease [59]:

- Albuminuria: too much of the protein albumine in the blood
- Hematuria: the presence of blood in the urine
- Reduced glomerular filtration rate: no efficient reduction of waste of the blood
- Hypoproteinemia: too little protein in your blood
- Edema: swelling due to an excess of body fluids

In all these cases it is advised to get a medical examination.

Albuminuria

Following the directive of the federation of medical specialists, there are three classifications of albuminuria: 1) Normal (A1), 2) Mildly increased (A2) and 3) Severely increased (A3). These classifications are based on the amount of albumine in the urine. The classification table is shown in table 1

	Morning urine albumine/creatine ratio (mg/mol)	**Morning urine albumine (mg/l)**	**24-hours urine albumine (mg/24 h)**
A1	<3	<20	<30
A2	3-30	20-200	30-300
A3	>30	>200	>300

Table 1: Classification albuminuria [70].

Hematuria

Hematuria is determined by means of a urine test with the help of a urine dipstick. With this test, the amount of erythrocytes (red blood cells) is measured. Furthermore, a urine sediment is taken, which will be analysed through a microscope. If there are more than 5-10 erythrocytes per ml and more than 3 erythrocytes per high power field, this can confirm hematuria [71].

Reduced glomerular filtration rate

The estimated glomerular filtration rate (eGFR) is an indicator for how fast the kidneys can filter waste out of the blood [72]. It is the amount of plasma water that passes the glomerular filters, per time unit. The directive of the federation of medical specialists describes 6 stages to classify the renal function, based on the eGFR: 1) normal (G1), 2) mildly decreased (G2), 3) mildly to moderately decreased (G3a), 4) moderately to severely decreased (G3b), 5) severely decreased and 6) kidney failure. The classification table for eGFR is shown in table 2

	eGFR (ml/min/1,73m^2)
G1	$\geq$ 90
G2	60-89
G3a	45-59
G3b	30-44
G4	15-29
G5	<15

Table 2: Classification eGFR [72].

Hypoproteinemia

Hypoproteinemia is determined by a significant loss in proteins - in particular albumin - because of disturbances in the synthesis. The amount of protein is indicated by g/dL [61]. The normal range of albumin is 3.4 to 5.4 g/dL. In case of a severe nephrotic syndrome, the level can drop down to 0.005 g/L [53].

Edema

Edema is often seen together with hypoproteinemia. Edema is the build up of fluid in a body, causing swelling [60]. The swelling often occurs in hands, ankles or the face. The function of the protein albumin is to hold water and salt inside the blood vessels. Because of the low level of protein, water leaks into the tissues, causing swelling [63].

2.1.1 Glomerular diseases

Research of Ayar et al. has shown that different factors are of influence when having a glomerular disease. Sex, age and geographical location are all of influence [4].

Glomerular diseases can be divided into primary and secondary glomerular diseases. The main difference is that primary diseases are not caused by a systematic disease such as diabetes, whereas secondary diseases are [60].

The most common glomerular diseases are listed in table 3, based on the research of Ayar et al.

Disease	**Explanation**
Minimal Change Disease (MCD)	Leaking of proteins in the filters of the kidneys [21]. Characterised by structurally normal glomeruli. May also occur in older adults who have nonspecific focal areas of tubulointerstitial scarring.
Focal Segmental Glomerulosclerosis (FSGS)	Developing scar tissue on the filter of the kidneys [13]. Characterised by sclerosis of a portion of the glomeruli.
IgA Nephropathy (IgAN)	Damage of the filters of the kidneys, caused by Immunoglobuline A (IgA) that is stuck in the filters [15]. Characterised by hematuria and varying proteinuria.
Lupus Nephritis (LN)	Forming of anitbodies against itself, which will attack the kidneys [19]. There are six classes indicating the severity of Lupus [27]

Table 3: Most common glomerular diseases [4]

.

However, there are a lot more glomerular diseases which can occur. Within the Radboud UMC, a list of the most occurring diagnoses is available. The list of all these diagnoses and their symptoms is attached in Appendix A.

2.1.2 Analysis

The analysis of glomerular diseases is usually performed by means of a blood test and/or urine test. When these tests do not make a grounded diagnosis, a biopsy can be taken. For a biopsy, a little tissue of the kidney is taken for further research. The tissue is placed under a microscopy and is analysed by pathologists [2]. On average, a pathologist in the Radboud UMC analyses 4 biopsies a day. An analysis of one biopsy take around 30 minutes.

The analysis of the tissue is done through histopathology. There are three microscopical examinations which help to form a diagnosis: 1) light microscopy, 2) immunofluorescence and 3) electron microscopy. Light microscopy is used to characterise abnormalities. Immunofluorescene is used to detect the presence of immunoglobulin and complements (proteins). Finally, electron microscopy is used to analyse the structure of the glomerular basement membrane (GBM), podocytes and depositions [58].

During the analysis, there are certain general characteristics the pathologists analyse, also depending on the kind of microscopy. In table 4, the characteristics that are used in a histopathological examination are described. A lot of these terms are commonly used in pathology reports.

Category	**Characteristics**
Generally descriptive	Focal: <80% damage to the glomeruli
	Diffuse: >80% damage to the glomeruli
	Segmental: lesion in some parts of the glomeruli
	Global: lesion in all parts of the glomeruli
Light microscopy	Normal glomerulus
	Non-proliferation: - sclerosis and hyalinosis - adhesion - hypertrophy - depositions - Abnormalities glomerular basement membrane (GBM)
	Proliferation: - mesangial (increase of mesangium cells) - endocapillary (increase of cells in capillaries) - mesangiocapillary (increase of cells in capillaries and in mesangium) - extracapillary (increase of cells in Bowman's capsule)
	Thrombosis in glomeruli or arterioles
Immunofluorescence	Localization of deposition
	Pattern of depositions: linear versus granular
	Type immunoglobuline: IgG, IgM, IgA, kappa and/or lambda chains
	Type complement factors: C3, C1q
Electron microscopy	Localisation of depositions: subendothelial , subepithelial or mesangial
	Aspect depositions: without structure (dense) or with structure (fibrillary, tubular, crystal)
	Aspect GBM: width, structure
	Aspect podocytes

Table 4: Characteristics histopathology [58].

2.2 Current situation

As aforementioned, within the Radboud UMC, an effort was made to digitalize the pathology reports. These reports are stored as one PDF file, where all the reports are included. For the department of nephrology there is already a text mining algorithm designed. The text off all the PDF files is extracted and stored within one text file. Because of privacy issues, all the pathology reports are anonymized. This means that patient names and personal details such as birth year are removed.

The text mining algorithm itself, is based on use cases of the Radboud UMC and therefore not published. An example is the difference in the level of the immunoglobulins kappa and lambda. Nephrologists suspect that a big difference in those immunoglobulins can indicate medical problems within the bone marrow.

The algorithm is designed in *Python* and makes use of regular expressions to capture meaningful information, such as the kappa and lambda levels, as mentioned above. Besides this algorithm, a user interface is designed to easily look up specific terms in the pathology reports. One can enter a term, and the user interface will show all reports with that specific term.

However, nephrologists from the Radboud UMC posed the question whether more detailed information could be extracted, such as a link between the pathology reports and the final diagnosis (or diagnoses). The nephrologists thus raised the question whether there is an efficient way to analyse these pathology reports by means of a text mining algorithm.

There are several benefits to analyzing pathology reports, which are discussed in section 1. Examples are linking a diagnosis (or diagnoses) to a pathology report and designing a classification model, which predicts the right diagnosis (or diagnoses) of the reports. This information can support nephrologists in their decision making process, as histological examination is a time consuming process and a lot of histological factors are of influence when stating a diagnosis. Thus when a nephrologist states their findings, a classifying text mining algorithm can state the corresponding diagnosis (or diagnoses).

One reason why classifying pathology reports was not possible before, is because there is no structured field within the pathology reports stating a clear diagnosis. Each pathology report has a section *conclusie* (conclusion), which describes, rather than states a diagnosis. Also, some pathologists explicitly state which glomerular disease is *not* diagnosed. This combination of descriptive diagnoses and counter-intuitive texts makes it not easy to extract the right diagnosis or diagnoses. As a result, it is also hard to form training- and test data sets.

Another reason, is that there is a knowledge gap between the two research fields. A nephrologist doesn't have knowledge about text mining and a bio-computer scientist doesn't have the full knowledge about nephrology. Because of this gap, the possibilities in analyzing these nephrology reports are not used to its full potential.

2.3 Text mining algorithms

Text mining is described by Feldman and Sanger as:

> "The process of extracting implicit knowledge from textual data." [25]

To give a more detailed definition, the description of text mining by Kumar et al. is included:

> "A process which transforms and substitutes (...) unstructured data into a structured one to facilitate knowledge extraction for decision support and deliver targeted information." [42]

A text mining algorithm thus reconstructs unstructured natural language into structured natural language. In particular, document classification is often used. This text mining algorithm task assigns a class (label) to a particular document. This is often done by classification algorithms, which will be elaborated on in section 2.3.2.

Text mining algorithms are often used in the healthcare sector in order to extract relations. A main problem is unstructured data. About 80% of the data used in the healthcare sector is unstructured [23]. This leads to challenges in designing a text mining algorithm, in particular during the preprocessing phase.

A text mining process generally has three main phases: 1) information retrieval, 2) information extraction and 3) knowledge discovery [69].

2.3.1 Information retrieval

Information retrieval is defined by Manning et al. as follows:

> "(...) finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers). [52].

This means that a user can enter a query into a system, to retrieve the right information. The information retrieval process often starts with *indexing* documents. This process creates an efficient representation of a document, to search for a document in a fast manner. An example is an *incidence matrix*. This binary matrix represents whether certain terms are included in the document. The terms (columns) are the indexed units [52]. The indexing process is schematically shown in figure 3.

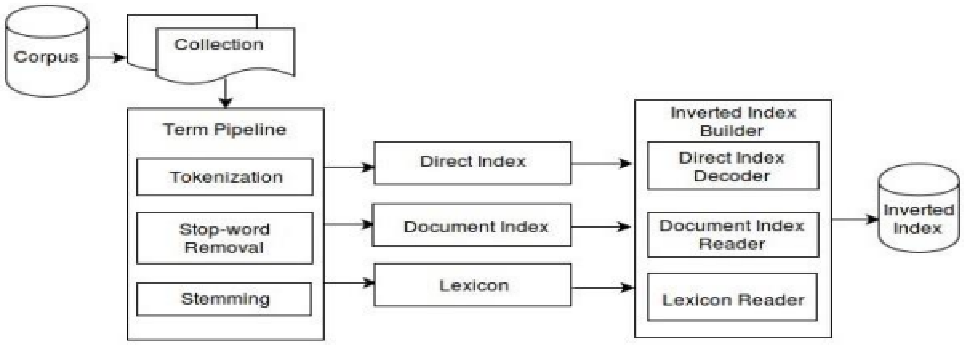


Figure 3: Indexing process [34].

As figure 3 shows, the collection of documents are first off going through a process called *term pipeline*. Within this process, three sub processes are executed: 1) tokenization, 2) stop-word removal and 3) stemming.

1) Tokenization

Tokenization is the process of chopping text up into pieces: *tokens*. This is done by segmenting the data by white spaces or punctuation [73].This process produces terms for the documents, which are included in the information retrieval system. It often chops up the normalized text into words. An example of tokenization is shown in figure 4.

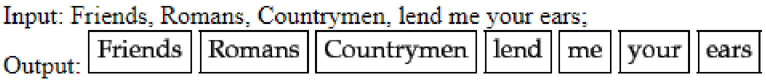

Input: Friends, Romans, Countrymen, lend me your ears;
Output: Friends | Romans | Countrymen | lend | me | your | ears

Figure 4: Example of tokenization [52].

2) Stop-word removal

Stop-word removal is the process of removing frequent words in the natural language. Examples of stop-words for the English language are ‘a’, ‘and’ and ‘or’. Removal of these stop-words can however lead to the lost of the context of the text.

3) Stemming

Stemming is the process of removing inflectional endings, by reforming words to their base words [34]. This process thus converts each token to its root form with grammar rules [73]. An example of stemming is:
`car’s, cars, car, cars’` → `car`.

A stemming algorithm that is often used, is *Porter's algorithm*, which sequentially performs word reductions [52].

After the term pipeline, an index is build. There are four main data structures for indexes [34]:

1. Direct index; stores the terms and their frequencies of a document
2. Document index; stores document relevant information, such as a document id and the length of a document in terms of the number of tokens
3. Lexicon; also stores the terms and their frequencies, but stores the global frequency
4. Inverted index; for each term, the corresponding documents and the term frequency is shown

With the inverted index, a query can be passed to this index, where several matches can be found. Thus, a fast overview of all the documents containing the terms in the query can be found. For a text mining classification algorithm, the input is mostly an incidence matrix or a direct index. However, using all possible words as features in an incidence matrix results in a time consuming process. To reduce the time of this process, feature selection is used.

Feature selection

Feature selection is the process of finding the smallest subset of features which is meaningful for your model and makes a classifier more efficient [52]. There are three techniques to perform feature selection: 1) Filter method, 2) Wrapper method and 3) Embedded method [74].

1) Filter method

The filter method selects a subset of the features based on inherent characteristics and thus independent of any learning algorithm (see figure 5).

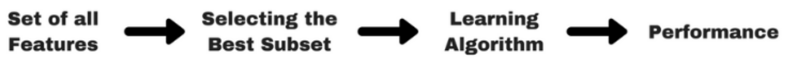


Figure 5: Filter technique feature selection [35].

The filter method that is used most often is the χ^2, to test the independence between the occurrence of the term and the occurrence of the class [52]. This method tests whether there is a significant difference (i.e. a p-value $\leq$ 0,05) between the observed and expected frequency of classes. If so, the words are not included in the set of features.

2) Wrapper method

The wrapper method selects a subset of features, based on the resulting performance of a classification algorithm (see figure 6) [73].

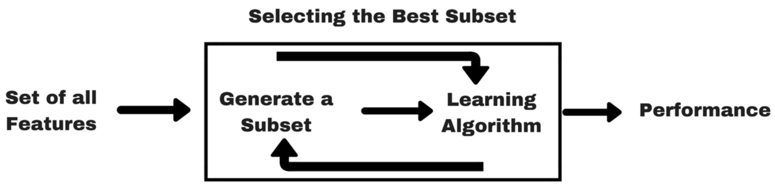


Figure 6: Wrapper technique feature selection [35].

The selection of the features can be done by *forward* or *backward* selection.

Forward selection starts with an empty set of features - a one dimensional vector - and adds new features, based on the best performance. The process of adding new features goes on until the performance of the classification algorithm improves [73].

Backward selection is the opposite of forward selection and starts with a set of all features. One by one, a feature is deleted from the set, based on an increase in performance after deleting it. This process of deleting features goes on until there is no increase anymore in performance after deleting a feature [73].

Just like the filter technique for feature selection, the wrapper method has also a statistical background in selecting the features. The features with a significant difference between the expected and observed frequency are not included in the set. This could be done by either selecting the features with no significant differences (forward selection) or deleting the ones that do have a significant difference (backward elimination).

3) Embedded method

The embedded method selects a subset of features, based on fitting the model and performing feature selection at the same time and thus with intervention of a classification algorithm (see figure 7).

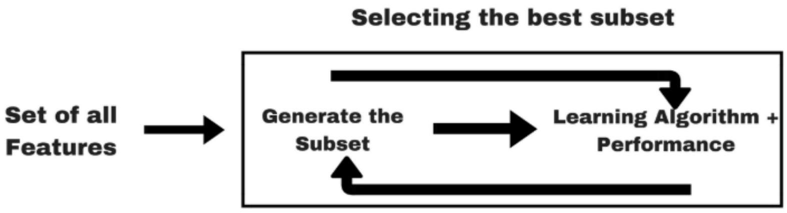


Figure 7: Embedded technique feature selection [35].

2.3.2 Information extraction

A method that is often used to perform information extraction is classification. Classification is described by Tan et al. as follows:

> "(..) the task of learning a target function f that maps each attribute set $\mathbf{x}$ to one of the predefined class labels y. [74]"

Classification is a form of a predictive model task, where the goal is to build a model for the target variables as a function of the explanatory variables [74]. The text mining of pathology reports requires mostly supervised data classification. With supervised data classification, new objects are classified, based on objects with a known class label [74]. Hence, the supervised data are the objects with a known class label. This approach is done because a specific target - the diagnosis - is to be predicted. This is easier when there is already data available with a known diagnosis.

The main similarity between a lot of text mining algorithms, is the use of a training- test and validation data set. The training data set is used to fit the text mining model. Thus, this is the actual data that is used to train the model. When a model is trained, the test data is used to see whether the model makes the right decisions, based on unseen data.

The use of a validation set is also often used. This approach divides the original training set in two subsets. One of these subsets is used as training data, whilst the other is used for validation: estimating the generalization error [74]. The validation set is often used to tune parameters for a model, for example to determine the best depth of a decision tree classifier. It in this way, the test data is held back only for the purpose of testing the model on unseen data. When a validation set is not used, the test- or training set is sometimes used to tune parameters for the model. However, this is perceived as "peeking", as the parameters are specifically tuned for data the model already knows [65].

Binary versus multiclass versus multilabel

A classification algorithm can either be a *binary*, a *multiclass* or a *multilabel* algorithm.

A binary classification algorithm focuses on binary classification: a label is either present (1) or not (0). Hence the name "binary", as there are only two options. However, often there are more than two categories. In this case, it is called multiclass classification.

A multiclass classification can be handled in two ways. The first way is to split the problem up in N binary problems. This means that with 8 classes, there are 8 binary problems. Another way is to set up a voting scheme. There are set up $N(N-1)/2$ binary classifiers, where each classifier distinguishes a sample between a pair of classes. Often, with majority vote, the final classification is determined [74].

Finally, there is a multilabel algorithm, which focuses on assigning a set of target labels to a sample. For example, when there are 5 classes, the set of target labels would be for example [0,0,0,1,1]. The classifier predicts in this case, that only the fourth and fifth class are present [45].

Thus the main difference between multiclass and multilabel classification, is that with multiclass classification, the classes are mutually exclusive, whereas with multilabel classification they are not.

Classifiers

Within the literature, there are various classifiers which are used. A decision tree classifier and neural network classifier are often used within clinical decision support, healthcare administration and text mining [31] [75].

Decision tree classifier

A decision tree classifier builds up a *decision tree*. This decision tree consists of nodes with test questions. For each test question, a node is splitted into two nodes. This process is repeated until one arrives at a leaf node, where the class label of the target can be found, based on these test questions. An example of an decision tree is shown in figure 8.

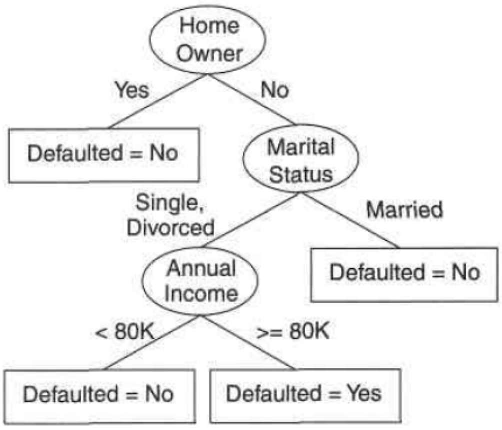


Figure 8: Example of an decision tree [74].

For this classifier, there are different algorithms which serve the purpose. The most used algorithm is *Hunt's algorithm* [74]. With certain test conditions, the attributes to split on are chosen. This turns the data set into purer subsets of the data. The splits are often the distinct values of that attribute (e.g. for the attribute sex, the distinct values are "male" and "female"). The test condition is mostly based on the gain of information the algorithms gets by splitting that certain attribute. By applying this recursively, the data set will be optimally pure at the end of the algorithm, where each subset of value(s) has a distinct label.

A limitation of the decision tree classifier is that it is very sensitive to small perturbations in the data and overfitting. Furthermore, it has problems with out-of-sample predictions: data that is not in the sample when fitting a decision tree classifier [41].

Neural Network

A neural network (NN) is often seen as a black box, where only the input and output matters. Partly because there is quite a complex mathematical background for this algorithm [5]. A neural network has a biological background, where it is based on the neural networks of the brains.

A neural network thus comes from a biological background, where neurons (nerve cells) need a stimulus (input) to perform an action (output) [74]. The network is often made out of three components: the input layers, the hidden layer(s) and the output layer(s) as shown in figure 9.

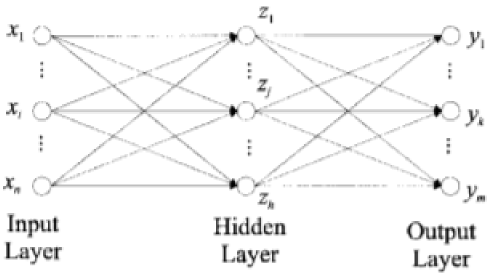


Figure 9: Multilayer neural network [5].

The neural network is established through a so-called *weight-function* that focuses on the error the model makes when using a training set. During the training process of the neural network, the weights are adapted, such that they fit the input-output relationships of the data [74]. These processes all happens within the hidden layers and is often also why the NN is called a black box: "hidden" relations are captured by a NN [5]. After this, a new unknown record can be used as input, where the output is the label of the record. An advantage of a neural network is that it can infer unseen relationships on unseen data, as it detects all possible interactions with the particular labels [76]. However - just as the decision tree classifier -, the neural network classifier is prone to overfitting.

TensorFlow

TensorFlow - designed by Google - is an open source library. It can be used to train neural networks for different purposes, such as image recognition, hand writing recognition and word embedding. TensorFlow enables users to graphically see the data flow through a graph. TensorFlow makes use of a *tensor*: a multidimensional array, which is categorized. This categorization is based on the order of the data. For example, a scalar is a order-zero tensor, a vector a order-one tensor and a matrix a order-two tensor. This can be graphically shown through nodes, where the "legs" of the nodes denote the order. These legs also have a dimension, which indicates the size of that leg. For example, a vector which illustrates the speed of an object in space, would be a three-dimensional order-one vector [30]. In figure 10, a graphical representation of four tensors are shown.

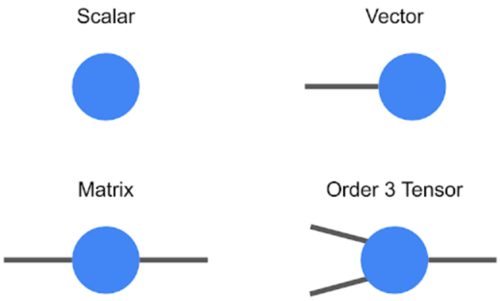


Figure 10: Graphical representations tensors [30].

With this representation, mathematical operations - and thus the data flow - can be encoded. Such an operation is called a tensor contraction: a mathematical summing operation which reduces the tensor rank [82]. In figure 11, there is an example of three order-three tensors, which are contracted. Furthermore, there are three dangling legs, which indicates the order of the resultant tensor. In this case, the remaining tensor would be a order-three tensor [30].

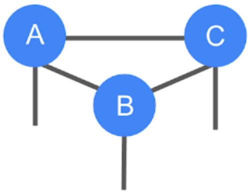


Figure 11: Tensor contraction [30].

The tensor contractions leads to efficiency in the mathematical operations in the neural network and therefore performing faster.

Ensemble methods

When using a certain classifier, ensemble methods can be used in order to improve the performance of a classifier. An ensemble methods forms a set of base classifiers and classifies samples by the combination of the prediction of each base classifier [74]. There are different methods to construct such a ensemble method, such as manipulating the training set or features. Below, often used ensemble methods are described.

Boosting

Boosting is an ensemble method, which manipulates the training set. This method iteratively changes the distribution of the training examples in different rounds, such that the classifier focuses on samples that are hard to classify [74]. Each sample from the training set is fitted with the particular classifier, such that the combination of the predictions is more accurate than just a single prediction [66].

A popular boosting algorithm is *AdaBoost*. This algorithm repeatedly takes a base learning algorithm, whilst maintaining a distribution, or set of weights over the training set. Initially, the weight of the training samples are set equally. In each round of taking a sample, the weights of incorrectly classifier samples are increased, such that the base classifier focuses more on the rare examples in the training set [66]. Because of the focus of specific samples, boosting is more prone to overfitting.

Bagging

Bagging is an ensemble method, which just like boosting, manipulates the training set. This method repeatedly takes samples from the training set, with replacement [74]. This means that some samples are in more than one training set, whilst others are in none of the training set. On average, a bootstrap sample contains approximately 63% of the original training set. The goal of bagging is to reduce the variance of the base classifier. Also the robustness of the classifier plays a role. The more unstable the base classifier is, the best bagging helps to reduce the errors. Because bagging does not focus on specific samples - like boosting does -, it is less susceptible to model overfitting when applied to noisy data [74].

Random Forest

The random forest classifier is used in combination with a decision tree classifier and combines the results of multiple decision trees. Each tree is generated, based on the selection of random vectors. The random forest classifier introduces randomness in order to minimize the average correlation between the trees [6]. The building process of a random forest is shown in figure 12.

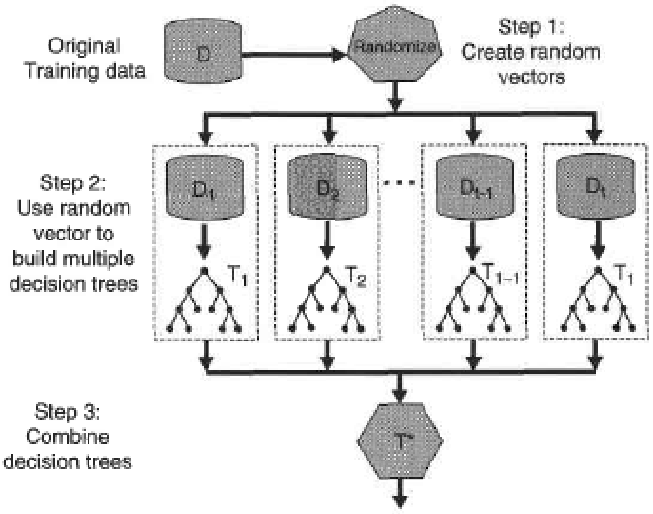

Original Training data
D
Randomize
Step 1: Create random vectors
Step 2: Use random vector to build multiple decision trees
D_1
D_2
D_{t-1}
D_t
T_1
T_2
T_{t-1}
T_t
Step 3: Combine decision trees
T*

Figure 12: Random Forest [74].

Class imbalance
When a dataset is imbalanced, there is undesirable class imbalance. Fortunately, there are certain techniques to handle this.

One technique is *oversampling*: increasing samples of minority classes, until there is an equal number of positive and negative examples [74]. A popular used algorithm for handling minority classes is *SMOTE*: Synthetic Minority Over-sampling Technique. This technique makes "synthetic" examples, based on existing samples in the dataset [24]. Rather than oversampling with replacement, new examples are made based on the k nearest neighbors of a minority sample.

Another technique is *undersampling*: decreasing samples of majority classes, until there is an equal number of positive and negative examples [74]. This can be done by random or focused subsampling of the data set.

2.4 Knowledge discovery

In this process, actual new information is extracted out of the natural language. For example, the classification of a pathology report leads to the knowledge of the diagnosis of the particular report. To evaluate the performance of a classification, certain model evaluation methods can be used.

2.4.1 Model evaluation method

Below, the three often used model evaluation methods are elaborated.

Holdout method
The holdout method divides the data into a training and test set. Often, 1/3 of the data is used for testing and 2/3 for training [74]. This method has several limitations, such as that certain samples are not included in the training set, but are in the testing set. This results in a sub optimal fitting phase, where predictions can be less precise.

K-fold cross validation
K-fold cross validation (hereafter called 'k-fold') is widely used to determine the skill of models and determine the predictive capability of a model. The algorithm splits the data in k groups (also referred to as folds). One of these groups forms the test data, whereas the rest of the groups forms the training data. This process of forming test- and training data repeats itself until all the groups are used for test data once [74]. The advantage of k-fold over the holdout method, is that k-fold is trained on more than one train-test data combination, which gives a more precise indication of the performance.

Often, a specific type of k-fold, called *stratified k-fold* is used for model evaluation. Stratified sampling means that whilst sampling, an equal amount of objects are picked from the group, even if they are imbalanced [74]. This means that the percentages of the samples for each class are maintained throughout the folds.

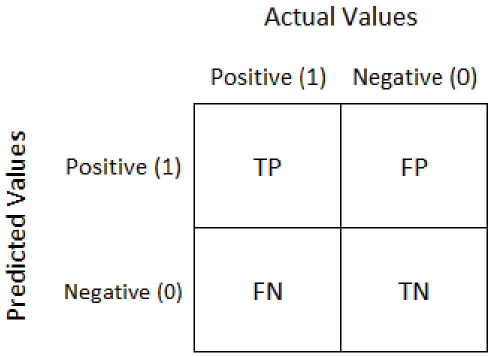
Bootstrap

The bootstrap method generates - just like k-fold - training and test data repeatedly in different runs. The difference is that with the bootstrap method, it is done with replacement. This means when a sample is chosen for the training set, it can be chosen again in the next run. The sampling is repeated b times. On average, a bootstrap sample contains $\pm 63,2\%$ of the original data [74].

2.4.2 Comparing classifiers

After the model evaluation, classifiers can be compared with the use of model evaluation metrics. These evaluation metrics are often based on a so-called *confusion matrix* (see figure 13). This matrix shows the amount of samples that are correctly or incorrectly classified, based on binary classification. A confusion matrix holds the following values:

- True positive (TP); the amount of positive samples that are rightly predicted to be positive by the classifier
- False positive (FP); the amount of negative samples that are falsely predicted to be positive by the classifier
- False negative (FN); the amount of positive samples that are falsely predicted to be negative by the classifier
- True negative (TN); the amount of negative samples taht are rightly predicted to be negative by the classifier



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 13: Confusion matrix [56].

Below, three often used evaluation metrics are described, including their advantages and disadvantages.

Accuracy

Accuracy is defined as follows:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad [74]$$

Accuracy focuses more on TP and TN, than on FP and FN. This can have a disadvantage in case of inbalanced data. For instance, when there is a data set with 100 samples, and just two positives. When a classifier predicts those 2 positive samples wrong (and thus FN), the accuracy would be $\frac{98}{100} = 98\%$. It thus seems like the classifier has a high performance, whilst the actual positive samples are predicted falsely to be negative.

F1 score

The F_1 score is defined as follows:

$$\mathrm{F}_1 = \frac{2 \times TP}{2 \times TP + FP + FN} \quad [74]$$

The F_1 score is build up from two other metrics: *precision* and *recall*. Precision describes the fraction of samples that is actually positive with regards to what the classifier predicted to be positive:

$$\text{Precision} = \frac{TP}{TP + FP} \quad [74]$$

Recall (also called *true positive rate*) describes the the fraction of positively samples that are correctly predicted by the classifier:

$$\text{Recall} = \frac{TP}{TP + FN} \quad [74]$$

.

The advantages of choosing F_1 measure above accuracy is thus that it is more suitable for inbalanced data, as it focuses more on FN and FP.

Receiver Operating Characteristic (ROC) curve

The ROC curve displays the tradeoff between the true positive rate (TPR) and the false positive rate (FPR). The TPR and FPR are defined as follows:

$$TPR = \frac{TP}{TP + FN} \quad [74]$$
$$FPR = \frac{FP}{FP + TN} \quad [74]$$

A ROC curve is plotted with the FPR shown on the x axis and the TPR shown on the y axis (see figure 14). In an ideal situation, the TPR is 1 and the FPR is 0. This would result in a straight corner. However, in practice, there is more of a curve as shown in figure 14.

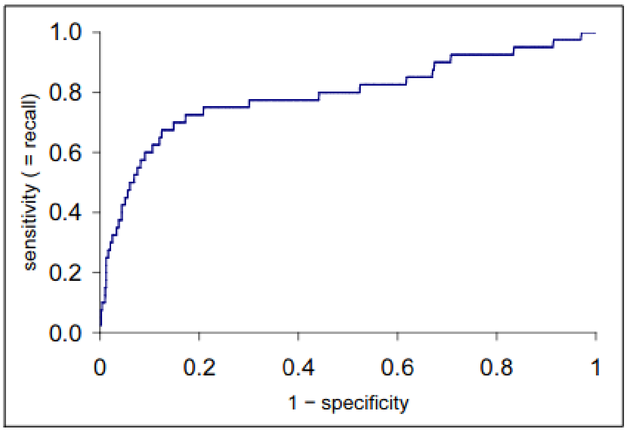


Figure 14: ROC curve [52].

2.4.3 Subconclusion

Within this theoretical framework, the following aspects are described: glomerular diseases, the current status of the pathology reports which are to be examined and the literature on text mining algorithms used in the healthcare sector.

It is clear that a lot of factors are of influence when performing histological examination (see table 4). Together with the fact that natural language - in particular medical notes - are often unstructured, this forms a challenge when performing classification tasks. For example, feature subset selection is more of a challenge, when there is no overlapping structure between the reports. This lack of overlapping structure could be of negative influence in performing feature subset selection. Furthermore, in the current situation, there is no clear diagnosis (or diagnoses) stated in the pathology reports. This means that it not yet possible to generate test data for a classification algorithm. This means that before a classification algorithm can be designed, a diagnosis (or diagnoses) has to be linked to each pathology report.

Classification (mapping attribute x to label y) can be done with the help of different classifiers. The most often used classifiers within the healthcare sector are the decision tree classifier and the neural network classifier.

An advantage of the decision tree classifier is that it quite self-explanatory. It is clear which choices the decision tree made to get to a specific label. However, a disadvantage of the decision tree classifier is that is prone to overfitting and performs worse on out-of-sample predictions. Thus, in case of rare glomerular diseases, the decision tree classifier could have problems with out-of-sample predictions.

An advantage of the neural network classifier is that with its weight-function, it can infer relationships in unseen data, because it detects all possible interactions with the particular labels. However, the neural network classifier is also prone to overfitting. Thus, overfitting could be a problem in combination with unstructured data when classifying the pathology reports.

With this new knowledge, subquestions one and two - as mentioned in the introduction - are answered.

3 Method

This section describes the method to classify the pathology reports. The method is subdivided into 5 parts: 1) Data collection , 2) Data pre-processing, 3) Data exploration and visualisation, 4) Model building and 5) Model evaluation. In figure 15, a schematic overview of the method is shown.

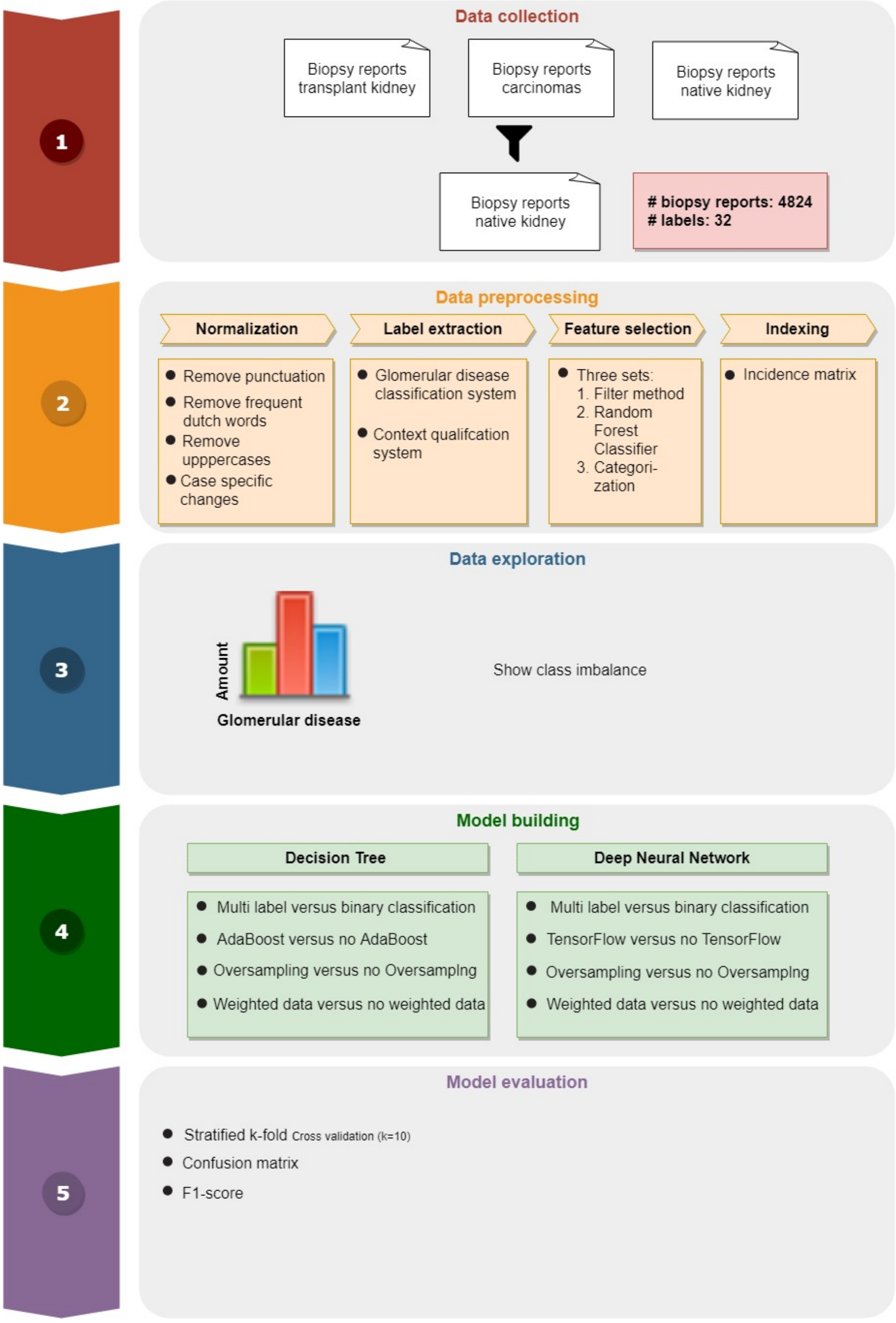

1
Data collection
Biopsy reports transplant kidney
Biopsy reports carcinomas
Biopsy reports native kidney
Biopsy reports native kidney
# biopsy reports: 4824
# labels: 32
2
Data preprocessing
Normalization
Label extraction
Feature selection
Indexing
Remove punctuation
Remove frequent dutch words
Remove upppercases
Case specific changes
Glomerular disease classification system
Context qualifcation system
Three sets:
1. Filter method
2. Random Forest Classifier
3. Categori-zation
Incidence matrix
3
Data exploration
Amount
Glomerular disease
Show class imbalance
4
Model building
Decision Tree
Deep Neural Network
Multi label versus binary classification
AdaBoost versus no AdaBoost
Oversampling versus no Oversamplng
Weighted data versus no weighted data
Multi label versus binary classification
TensorFlow versus no TensorFlow
Oversampling versus no Oversamplng
Weighted data versus no weighted data
5
Model evaluation
Stratified k-fold Cross validation (k=10)
Confusion matrix
F1-score

Figure 15: Schematic overview of the used method.

The used software for the text mining process is *Python 3.6*. Within *Python*, a lot of libraries are used to support this process. A list of the included libraries in Python is attached in Appendix B. Furthermore, the code is uploaded in Gitlab (see `https://gitlab.science.ru.nl/jvisschedijk/text-mining-pathology-reports`).

3.1 Data collection

The dataset exists of the pathology reports of the nephrology department of the Radboud UMC in Nijmegen. These reports are anonymized and stored within a text file.

There are pathology reports of three types of biopsies:

1. Biopsies of native kidneys
2. Biopsies of kidneys with carcinomas
3. Biopsies of transplant kidneys

Only the first category describes an actual diagnosis of a certain glomerular disease. Thus, just the first category is included in the data set.

The dataset exists of 4824 pathology reports (samples), each existing of natural language. Each report has a semi-structured layout. The schematic structure of a biopsy report is shown in figure 16.

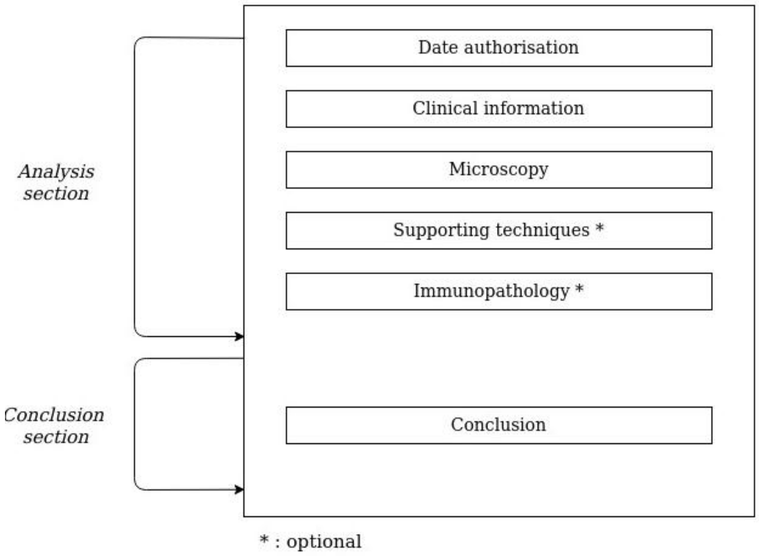


Figure 16: Layout pathology report

As shown in figure 16, there are several subsections which occur in every report, such as *microscopy* and *medical information*. The section which describes the actual diagnosis, is called *conclusie* (conclusion). All the text before the diagnosis section, is from now onwards referred to as the *analysis* section. Because of the fact that the conclusion section describes a diagnoses, a clear label is not (yet) present.

3.2 Pre-processing

The dataset is pre-processed with four procedures: 1) normalization, 2) label extraction, 3) feature selection and 4) indexing.

In the model building process, there are different models generated. Because of the different models, there are also slight differences in the pre processing of the data. These differences will be discussed below. The actual model building will be discussed in section 3.4.

1. Normalization

Normalization of the text is done by a combination of several basic normalization steps and some case specific steps. This procedure is done to prevent negative influences for the classify methods.

The basic normalization steps include:

- Removing upper cases
- Removing punctuation; the following punctuation is removed: !$'()*,.[]_;"/\
- Removing most frequent Dutch words; these includes mostly articles and catch phrases. The removed words are based on an online list of frequent Dutch words [62], and other frequent words occurring in the biopsy report. The full list of removed frequent Dutch words is attached in appendix C.

The case specific normalization steps include:

- Removing white spaces between the substring "1 +"; this substring refers to a level of certain proteins, such as IgA. However, sometimes nephrologists denote the level without the white space ("1+"). By normalizing this specific case, it is easier to analyse these levels if needed.
- Placing a white space when there is a question mark ('?'); the question mark is often placed after a diagnosis, indicating a doubting diagnosis. Because of the placement of an extra white space, it is seen as a separate word, to catch the doubting diagnoses.
- Removing dates and phone numbers.

2. Label extraction

As aforementioned, the conclusion section of a biopsy report describes one or more diagnoses, rather than clearly state them. Because of this, a clear label is not yet present. The first step to prepare the data for a classification algorithm, is to extract these labels.

A glomerular disease classification system is set up. For each glomerular disease, there is a unique identifier (*GDid*). Each *GDid* has a primary name and synonyms which can occur in the biopsy report. Both glomerular diseases as general glomerular abnormalities are included in the classification system. All the glomerular diseases of Appendix A are included.

In total, there are 31 different glomerular diseases. In figure 17, an example of the glomerular disease *fsgs* in the disease classification system is shown. The full glomerular disease classification system is attached in appendix C.

`GDid_06`	*name*	fsgs
`GDid_06`	*synonym*	focale glomerulosclerose
`GDid_06`	*synonym*	focale glomeruloscleroses
`GDid_06`	*synonym*	focale segmentale glomeruloscleroses
`GDid_06`	*synonym*	focale segmentale glomerulosclerose

Figure 17: Glomerular disease classification system. First column: identifier glomerular disease, second column: categorization of the name, third column: name (or synonym) of the disease

With this glomerular disease classification system, the "hits" can be found in the conclusion section. A hit can be described as the presence of the primary name or one of the synonyms of a glomerular disease in the conclusion section. When there is no hit, the particular record was labeled as "other". No hit can be described as the absent of one of the glomerular diseases, based on the glomerular disease classification system. The reason for this can be either that a particular glomerular abnormality is not in the glomerular disease classification system, or that there is no glomerular disease at all. With the extra label "other", there is a total of 32 different labels.

When there is a hit, the context is qualified through a set up context qualification system. This rule-based approached systems gives an numeric value (index) to certain words, indicating uncertainty or doubt. For example the word "*geen*" ("no") is an indicator for uncertainty. In figure 18 the context qualification system is shown.

no	0	uncertain
no classifying	0	uncertain
not	0	uncertain
insufficient	0	uncertain
negative	0	uncertain
?	1	doubt
maybe	1	doubt
perhaps	1	doubt
not convincing	1	doubt

Figure 18: Context qualification system. First column: word to be indexed, second column: index of certainty, third column: explanation of index value

Based on the indices of these words, a diagnosis can be qualified with either a 0 (= uncertain), 1 (= doubt) or 2 (= certain). Thus, for each hit, the context is qualified. The context is besides the qualification system also assessed by other (case specific) rules. Only the hits with a qualification of either 1 or 2 are included in the data set.

For 300 reports, the extracted labels of the biopsy reports are manually checked by the researcher and a bio-computer scientist, experienced in the field of text mining. This resulted in a 100% confusion matrix and thus an F-score of 1. The results of this label extraction process are further elaborated in section 4.

3. Feature selection

Feature selection is done with three different methods, to obtain three different set of features:

1. Filter feature subset selection
2. Random Forest classifier feature subset selection
3. Categorization

The feature selection procedure is executed in order to reduce both overfitting and the training time. Within the model building process, there are two classification methods used: multi label classification and binary classification. The first method is when a set of target labels is assigned to the biopsy reports. With the latter method, for each label (= each diagnosis), a separate classifier is used in order to predict whether the diagnosis is present or not. Because of the different classification methods, there is a difference in feature selection.

1. Filter feature subset selection (FFSS)

The feature subset selection is started off with tokenization of the analysis section of each report, where each token represents a feature. In total, there were 17141 feature tokens, after removing duplicate tokens.

Secondly, feature subset selection is executed to extract only the features that are actually interesting for a classifier. A filter method is used in order to retrieve the best features. With a χ^2 test, the best k features are selected, based on statistical independence. For each label, the 250 best feature types are chosen with the filter method called *SelectKBest* [49]. With 32 labels, this results in a total of 8000 feature tokens. After removing duplicates, the total amount of feature types is 5347. The process of the feature subset selection is schematically shown in figure 19.

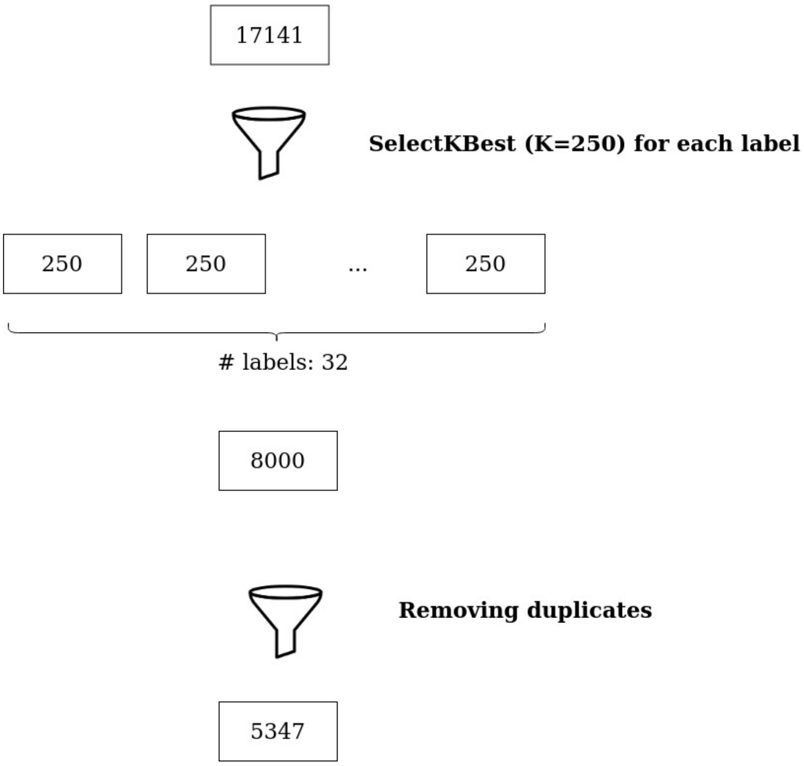


Figure 19: Feature subset selection with *SelectKBest* [49].

For multilabel classification, there is thus a total of 5347 feature types, based on the total of 8000 feature tokens.

For binary classification, there is a total of 250 feature types for each label.

2. Random Forest classifier feature subset selection (RFFSS)

The random forest classifier method is also started off with tokenization of the analysis section of each report, resulting in 17141 feature tokens. After this, features are extracted with a *Random Forest classifier* [47].

For this method, tokens which occurred in 75% of the reports were for each label collected for each. This resulted in 2632 feature types. An incidence matrix of these types is set up, which formed the input for the random forest classifier. For 100 runs, the incidence is fitted with the random forest classifier, with the following specific input parameters:

- max_depth = 90
- n_estimators = 100
- random_state = 0

The choice of these parameters are defined further in section 3.4.1.

Within each run, the importance of the features are extracted. Only the features with an importance equal or higher than the mean importance are selected [48]. This resulted in 68006 feature tokens, thus including duplicates.

In order to filter out the outliers, only the feature tokens that are present in 50% of the runs are recorded in the subset of final feature types. This resulted in a total amount of 305 feature types, after removing the duplicates. The process of random forest classifier feature subset selection is schematically shown in figure 20 .

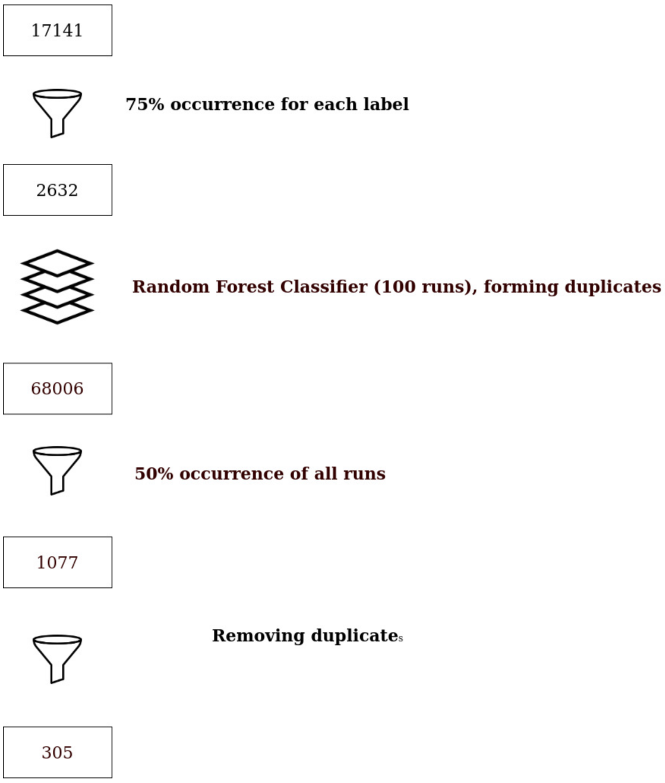

17141
75% occurrence for each label
2632
Random Forest Classifier (100 runs), forming duplicates
68006
50% occurrence of all runs
1077
Removing duplicates
305

Figure 20: Random Forest Feature Selection.

For multilabel classification, there are thus 305 features. However, for binary classification, there is a difference in the amount of features for each label, based on the importance of features. The amount of features for each label is described in Appendix C.

3. Categorization

Because not only certain words, but also specific features could be of importance, a categorization system is set up. This categorization system consists of 25 categories with pre-defined values. The categories are determined based on both certain features a pathologist uses when performing histological examination, as guidelines regarding glomerular diseases [58].

An example of a feature is the IgA level, for which the value is one of the following: '0+' , '1+', '2+', '3+', or 'trace'. The full categorization system is attached in appendix D.

These categories are transformed in integer numeric values as input for the classifiers. For example the IgA levels has five possible values, with corresponding values 1-5. This resulted in 25 features with a numeric values, based on the categorization system.

Both multilabel and binary classification has the same input matrix of 25 features. Because of the fact that the biopsy reports do not have a clear structure, the pre-defined values are hard to capture. The process of extracting the pre-defined values can thus be seen as a testing process.

4. Indexing
In the last step of the pre-processing procedure, the biopsy reports are indexed by means of an incidence matrix in case of the first two feature selection methods. For each feature, there is either a 0 (not present) or 1 (present) in the incidence matrix. This produces a binary representation of the presence of the features. For the last feature selection method, the categorization system, the corresponding numeric value of the category is recorded in the input matrix. As there are three sets of features, there are also three possible sets of incidence matrices for classification.

Final pre-processed data
The final pre-processed data is in the form of an incidence matrix ('X') with the corresponding labels ('y').

In case of multilabel classification, the labels are binarized, as one pathology report can have multiple diagnoses. As aforementioned, there is a total of 32 labels. This means that every pathology report has a binarized representation of the presence of all the 32 diagnoses. This results in a list of 0's and 1's representing the presence or absence of a diagnosis respectively.

In case of binary classification, there is also a binarized representation, but just for the particular diagnoses, i.e. a diagnosis is either present (1) or not (0).

3.3 Data exploration and visualisation

To give an overview of the available data, one visualisation is made. A bar plot showing the total amount of incidences of the glomerular diseases in the reports is generated. This to have clear overview of the possible class imbalance of the glomerular diseases.

3.4 Model building

As aforementioned, there are different models used, with different parameters.A schematic overview of the model building process is shown in figure 21.

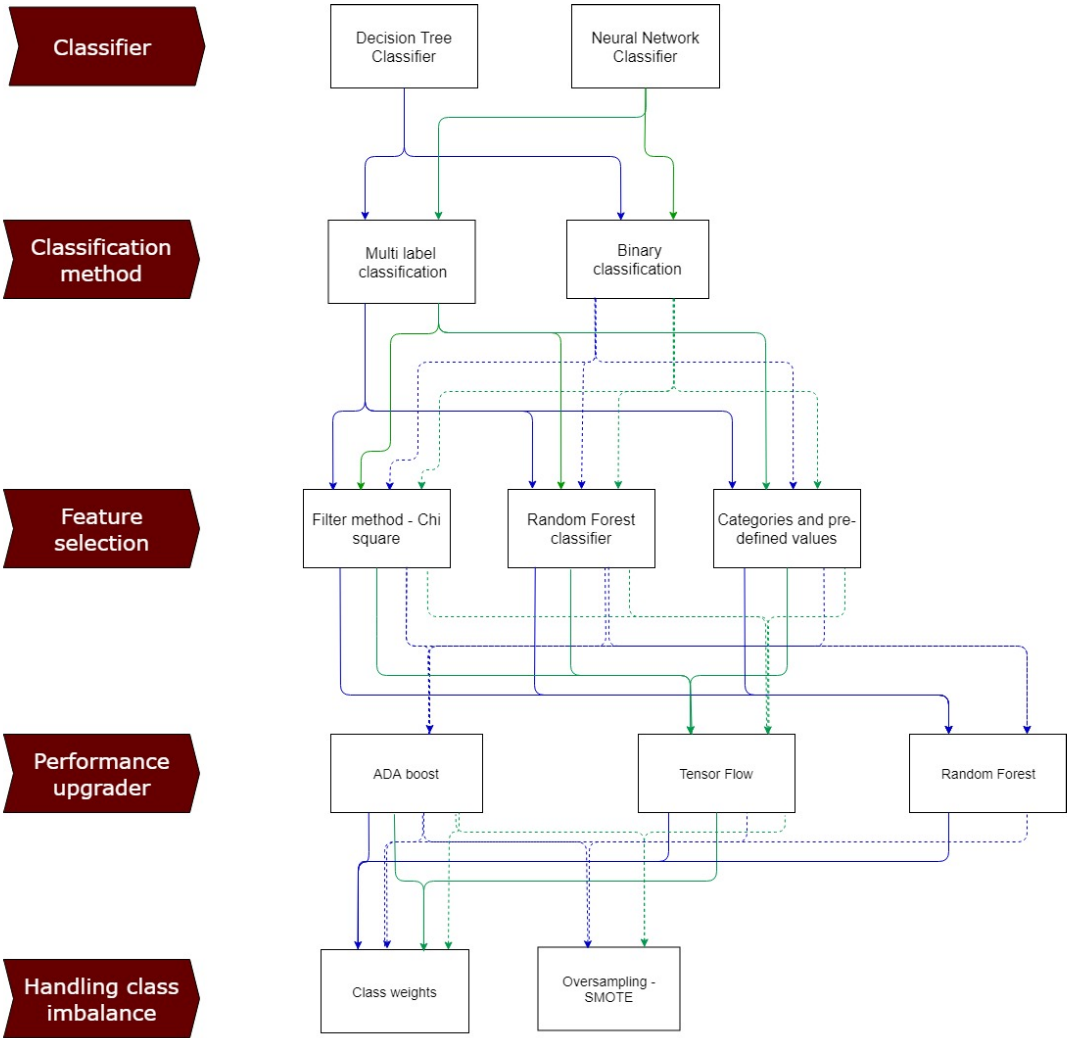


Figure 21: Model building process; Blue trace = Decision Tree classifier, Green trace = Neural Network classifier, solid trace = Multilabel classification, dotted trace = Binary classification.

As figure 21 shows, there are five categories: 1) Classifier, 2) Classification method, 3) Feature selection, 4) Performance upgrader and 5) Handling class imbalance. Note that from all the options within a category, just one of the options is chosen. For example, the following holds for the category "Classification method": *Multi label classification* $\oplus$ *Binary classification*. Furthermore, there are certain other restrictions, such as not using oversampling in combination with multilabel classification. This leads to a total of 63 models, of which 39 models use a decision tree classifier, and 24 models a neural network classifier. All the models with their parameters are described in appendix E.

In the sections below, the models will be discussed.

3.4.1 Classifier

There are two classifiers that are used to build models to classify the biopsy reports: 1) Decision tree classifier and 2) Neural network classifier. Research of Ardahapure et al. [3] and Fodeh et al. [26] have shown that the decision tree performs better than SVM, K-nearest neighbors and neural networks in terms of classifying medical documents. However, other authors claim otherwise and show that neural network classifiers outperform decision tree classifiers [54]. For this reason, these two classifiers have been chosen to use for the text mining algorithm.

For each of the models, the parameters for the model are chosen based on earlier studies and evaluation of the model using different values for the parameters.

1. Decision tree classifier classifier

The decision tree classifier is set up by means of the *Decision tree classifier* in Python [50].

The decision tree classifier has a maximum depth of 90, and a minimum split of 2, for both multilabel as binary classification. These parameters are based on a stratified k-fold (with k=10) evaluation with multilabel classification and filter feature subset selection, where the test and training errors are determined. For a varying depth between 10 and 100 with intermediate steps of 10, the test and training errors are determined. The depth with the minimum mean test error of 0.047 is chosen.

A research of Fodeh et al. used decision tree classifier in classifying clinical notes [26]. In this research, there was no pruning used, but model evaluation with stratified K-fold showed better results when setting a maximum depth.

2. Neural Network classifier

The Neural Network (NN) classifier is set up by means of the *Multilayer Perceptron* model in Python [44].

The NN classifier has 3 hidden layers, with each 64 nodes. The used parameters is based on both stratified k-fold (with k=10) multilabel evaluation with filter feature subset selection and research of Kwang et al. and Shah et al. In the first research, the authors used 10 hidden layers, whereas in the second research, the authors used a standard 3 layer model.

With stratified k-fold (with k=10), the right amount of hidden layers and neurons are determined, where the test and training errors are calculated. For an amount of hidden layers varying between 1 and 15, the errors are established with stratified k-fold. The amount of neurons for each hidden layer varied between 40 and 75, with intermediate steps of 5. The amount of hidden layers and amount of neurons with the minimum mean test error of 0.15 are chosen

3.4.2 Classification method

With the two aforementioned classifiers, models are build for the use of two types of classification methods: multi label classification and binary classification. For both the decision tree classifier, as the neural network classifier, a separate model is build with a multilabel and binary classification method (see figure 21). As certain glomerular diseases may appear secondary, multilabel classification is used in order to see any difference in performance as opposed to binary classification.

Multi label classification
For the multi label classification, the labels are binarized, as mentioned in section 3.2.

Binary classification
For this type of classification, there is a classifier built for each label, thus 32 in total. The differences in the data preprocessing for each label is described in section 3.2.

3.4.3 Feature selection

Feature selection is strictly speaking part of data preprocessing and thus is discussed in section 3.2.

There are different features for multilabel and binary classification. In binary classification, there is - besides the categorization system - a separate set of features for each glomerular diseases. For multilabel classification, those features are combined, resulting in more overlapping features.

3.4.4 Performance upgrader

In order to improve the performance of the two classifiers, three different methods are used: 1) AdaBoost (binary classification), 2) Tensor Flow (binary and multilabel classification) and 3) Random Forest (binary and multilabel classification).

As shown in figure 21, AdaBoost and Random Forest are only used in models which make use of a decision tree classifier. Tensor Flow is only used in models which make use of a neural network classifier.

AdaBoost
AdaBoost - used for binary classification - is used for the decision tree classifier, by means of an *AdaBoost classifier* in Python [46] . AdaBoost is a boosting classifier, which focuses on rare samples. As the dataset contains glomerular diseases which occur less than 10 times, boosting is a suitable option.

For the AdaBoost classifier, the base classifier is the decision tree classifier as described in section 3.4.1. The maximum amount of estimators is 200.

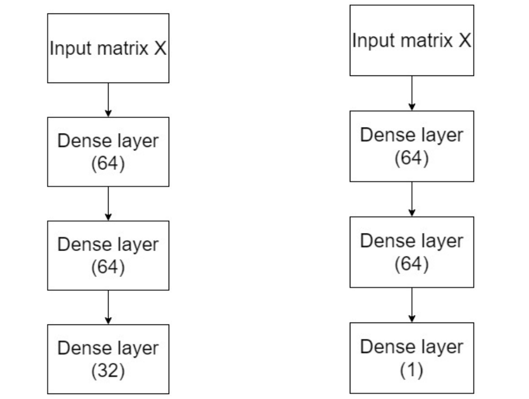
Tensor Flow

Tensor Flow - used for both binary and multilabel classification - is used for the neural network classifier, by means of a tensorflow backend [37]. This classifier has also 3 hidden layers, of which the first two have 64 node each. The nodes are fully connected, with 'relu' as activation function. The last layer has 'sigmoid' as activation function [36]. Tensor Flow is often used with text classification, using word embedding. However, as this is a very time consuming process and existing databases for word embedding are often in English, word embedding is not used. However, as Tensor Flow is often used for text based application, it is used to see whether it improves performance of the classifiers.

In case of multilabel classification, the last layer consists of 32 neurons. In case of binary classification, the last layer consists of a single neuron.

The classifier is compiled with 'adam' as optimizer [39] and 'binary cross entropy' as loss function [38]. After compiling the classifier, the classifier is fitted with 30 epochs, with a batch size of 3.

The schematic visualization of the classifier is shown in figure 22.



(a) Multilabel classification

(b) Binary classification

Figure 22: Structure Neural Network, using Tensor Flow.

The used parameters for a neural network with a tensorflow backend are based on both stratified k-fold evalutation (as aforementioned) and a research of Rajput et al. [64]. This research also used 'adam' as optimizer and 'binary cross entropy' as loss function. The authors used 6 layers, with different characteristics for their model, such as an embedded layer. However, as within this research, there is no use of word embedding, this layer was not suitable to use.

Random Forest

Random Forest - used for both binary and multilabel classification - is used with the decision tree classifier as base classifier. It is set up with a *Random forest* *classifier* in Python [47]. The maximum amount of estimators is 200. As random forest combines several decision trees, it could be improve the performance, as a more weighed decision is made.

3.4.5 Handling class imbalance

As some glomerular diseases are rare, and others occur in almost every biopsy report, this class imbalance needs to be handled. This is done in two ways: class weights (multilabel classification) and oversampling (binary and multilabel classification).

Class weights
Class weights can be used to express a certain importance to a class. With a class weight, certain classes can be emphasized, rather than just take the frequency into account.The class weight is computed as the relative occurrence of a label:

$$\frac{\#\text{samples training set} - \#\text{label}}{\#\text{samples training set}}$$

Oversampling
Oversampling is used to correct for the imbalanced data. For oversampling, SMOTE is used, in order to not just copy existing samples, but rather make synthetic samples, based on the k neighbors of that sample. A k of 3 neighbors is used to generate these synthetic samples. Because of this, it forces classifier to handle minority classes as more general classes.

3.5 Model evaluation

The model evaluation is performed with stratified k-fold cross validation, with $k = 10$. Stratified k-fold is preferred over normal k-fold cross validation, as with stratified k-fold, the folds preserve the percentages of samples for each class. A value of $k = 10$ is chosen, because it is proven that the test error rate does not suffer from high biases or high variances [32].

With this model evaluation, a test- and training set is used. The reason for not using a validation set, is because the total amount of samples is quite small. When this data would have been splitted into three sets (training, validation, test), this could result in less information gain for the classifiers and thus less performance.

For the models using multilabel classification, an iterative stratification approach for model evaluation is used. This approach is chosen above a label set approach, as the latter approach is very time consuming.

In each run of stratified k-fold, the training data is fitted, after which the test data is predicted. Of these predicted results, a confusion matrix is calculated. All the separate confusion matrix of each run, are then added to form one confusion matrix for each label. The final results of the second evaluation are thus 32 confusion matrices, one for each label.

Based on these confusion matrices, the F1 score of all the labels of the 63 models are calculated, in order to compare the models. As there is imbalanced data, it is important to take both precision and recall into account. Otherwise, a high accuracy could imply a good performance, whilst it could be that a classifier predicts all TN's right, but all TP's incorrect. Thus, the F1 score is a good indicated for performance within this thesis.

The comparison of the models is done by a bottom-up approach, where the difference in performances will be discussed based on the categories shown in figure 21.

4 Results

The results are structured in the same way as the processes described in the method section. First off, the results of the data preprocessing will be described. Secondly, the data exploration phase will be described. Finally, the results of classifying the biopsy reports with the decision tree classifier and deep neural network classifier are elaborated.

4.1 Data preprocessing

As described in the method section, a label extraction process has been executed in order to extract the right labels for each biopsy report. The first run of the label extraction process led to a total amount of 4957 biopsy reports, of which 77% had one or labels. Of these biopsy reports, 100 reports were manually checked to see whether the extracted labels were actually correct. Of the 100 checked reports, 50 had one or more label. The other 50 reports had no label at all, and thus should have no description of a glomerular disease.

The first run of manually checking the results led to the following confusion matrix:

	Actual		**total**
Prediction	48 (TP)	2 (FP)	
	13 (FN)	37(TN)	
Total	50	50	

After adding several glomerular diseases and performing other fine tuning, a second and final run of the label extraction process was executed. This time there was a total of 4824 biopsy reports, of which 92% had one or more labels. Of the reports, now 300 biopsy reports were manually checked. The manual check included 250 reports with one or more labels and 50 reports with no label. This led to a 100% correct confusion matrix, with 250 true positives and 50 true negatives.

Furthermore, some general findings can be described.

First off, the pathology reports are very unstructured. Due to the fact that these reports are written by different pathologists, different writing styles are used. Thus in order to pre-process the data and find relations or structure, is very challenging.

Secondly, it appears that some glomerular diseases often occur secondary to another glomerular disease. For example, focal segmental glomerulosclerosis (fsgs), often occurs secondary to IgA nephropathy and membranous nephropathy. Research has been conducted into this phenomenon, which describes the combination of the two glomerular diseases as a poor clinical outcomes [55] [79]. Also tubule interstitial nephritis is a secondary glomerular abnormality, as it occurs in a lot of pathology reports.

4.2 Data exploration and visualisation

The distribution of the amount of occurrences of the glomerular diseases is shown in the barplot in 23.

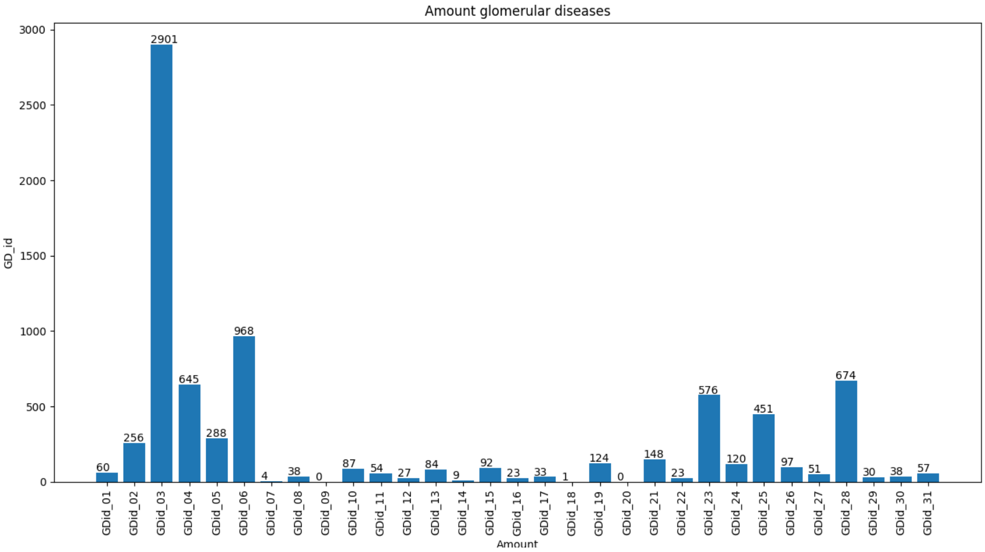


Figure 23: Amount glomerular diseases in the pathology reports.

As figure 23 shows, there is quite a class imbalance. The two glomerual diseases "MPGN Type III" (*GDid_09*) and "HCD" (*GDid_20*) are not present at all in the biopsy reports. Furthermore, with an amount of 2901, the glomerular disease Tubule interstitial nephritis (*GDid_03*) is highly over represented in the pathology reports. Finally, there are five glomerular diseases that are occur less than 25 times in the pathology reports. This imbalanced data can have high influence on the performance of a classifier. For example, because the glomerular disease Tubule interstitial nephritis is highly over represented, there is a high chance that a classifier will often predict that this disease is present.

4.3 Model evaluation

As aforementioned, there are two classifiers which are being reviewed: a decision tree classifier and a neural network classifier. In total, there are 63 possible models, with different parameters. The model evaluation for each model is based on stratified k-fold (k=10), with a test set of ± 482 samples and training set of ±4346 for each run.

Below, the results for the 5 best performing models for each classifier are described. The total model evaluation of all the models is described in appendix F.

4.3.1 Decision tree classifier

The decision tree classifier has the best performance with the following five models:

1. (a) Model number: #13
   (b) Classification method : Binary
   (c) Feature selection : Filter feature selection
   (d) Performance upgrader : None
   (e) Handling class imbalance : None
2. (a) Model number: #14
   (b) Classification method : Binary
   (c) Feature selection : Random Forest Features Selection
   (d) Performance upgrader : None
   (e) Handling class imbalance : None
3. (a) Model number: #22
   (b) Classification method : Binary
   (c) Feature selection : Filter feature selection
   (d) Performance upgrader : AdaBoost
   (e) Handling class imbalance : None
4. (a) Model number: #23
   (b) Classification method : Binary
   (c) Feature selection : Random Forest Feature selection
   (d) Performance upgrader : AdaBoost ‘
   (e) Handling class imbalance : None
5. (a) Model number: #37
   (b) Classification method : Binary
   (c) Feature selection : Random Forest Feature Selection
   (d) Performance upgrader : Random Forest
   (e) Handling class imbalance : Oversampling

The F1 score of these five models are shown below in table 5

Disease	**#13**	**#14**	**#22**	**#23**	**#37**
Acute glomerulonephritis	0.0487	0.1215	0.0267	0.0126	0.0153
Lupus nephritis	0.7315	0.7417	0.7194	0.7218	0.7635
Tubule-Interstitial nephritis	0.7206	0.6942	0.7279	0.6974	0.7958
IgA nephropathy	0.6028	0.3809	0.6027	0.3944	0.6614
Mcd	0.5675	0.4237	0	0	0.6731
Fsgs	0.5944	0.4599	0.5993	0.4479	0.6523
Mpgn Type I	0	0	0	0	0
Mpgn Type II	0.3667	0.3505	0.3279	0.3478	0.1584
Mpgn Type III	-	-	-	-	-
Hsp	0.6456	0.5856	0.5694	0.5484	0.3178
Fibrillary glomerulonephritis	0.7475	0.6415	0.7647	0.6852	0.7885
Amyloidosis AA	0.2667	0.421	0.3111	0.5	0
Amyloidosis AL	0.5882	0.497	0.5695	0.5767	0.6989
Immunotactoid glomerulopathy	0	0.2609	0	0.2353	
Pauci-immune vasculitis	0.1351	0.1053	0.0351	0.1138	0.0575
Anti-gbm nephritis	0.1714	0.0816	0.1143	0.1154	0
Postinfectious glomerulonephritis	0.0755	0.0784	0.08	0.0784	0.0128
Lcdd	0.5158	0.4511	0.5182	0.4428	0.4606
Hcdd	-	-	-	-	-
Diabetic nephropathy	0.4241	0.1514	0.4394	0.134	0.5758
Hereditary nephritis	0.1935	0.25	0.0645	0.1765	0
Lhcdd	0	0	0	0	0
Membranous nephropathy	0.8	0.7199	0.8068	0.7435	0.8818
Mesangiocapillary glomerulonephritis	0.5286	0.4528	0.8068	0.7435	0.5321

Disease	**#13**	**#14**	**#22**	**#23**	**#37**
Anca glomerulonephritis	0.6205	0.4796	0.6275	0.5126	0.694
Acute tubular necrosis	0.2899	0.0677	0.2154	0.0772	0.0906
Cast nephropathy	0.439	0.2927	0.4198	0.2933	0.0604
Tubulopathy	0.3338	0.2098	0.332	0.2223	0.2119
C3 glomerulopathy	0.4286	0.2564	0.2381	0.3014	0
Amyloidosis (No type)	0.1159	0.1584	0.1212	0.2385	0.0287
Focal segmental sclerosing glomerulopathy	0.0377	0.0131	0.0208	0.0144	0
Other	0.3313	0.162	0.3295	0.1693	0.46
Mean	0.3774	0.317	0.3530	0.3259	0.3197

Table 5: Model evaluation decision tree classifier; test data = ± 482 samples, training data = ± 4346 samples.

As table 5 shows, the mean F1 score lies around ±0.33. This means that on average, there are a lot of FN's and FP's in the predicted glomerular diseases. Furthermore, certain glomerular diseases have a fairly high F1 score. For example, the glomerular disease lupus nephritis has a score of ±0.73. When looking at the F1 score of the glomerular diseases, there is a relation between the amount of diseases and the F1 score. Glomerular diseases with an occurrence equal or higher than 500 in the pathology reports, have a F1 score around ±0.5 or higher. Rare diseases, such like MPGN Type I which occurs just four times in the pathology reports, is never predicted correctly.

Below in table 6, the F1 scores of the top five most occurring glomerular diseases (see figure 23) are shown.

Disease	**#43**	**#44**	**#46**	**#47**	**#49**
Tubule-Interstitial nephritis	0.7206	0.6942	0.7279	0.6974	0.7958
IgA nephropathy	0.6028	0.3809	0.6027	0.3944	0.6614
Fsgs	0.5944	0.4599	0.5993	0.4479	0.6523
Membranous nephropathy	0.8	0.7199	0.8068	0.7435	0.8818
Tubulopathy	0.3338	0.2098	0.332	0.2223	0.2119

Table 6: F1 scores for top 5 most occurring glomerular diseases.

As table 6 shows, the top five most occurring glomerular diseases have - besides tubulopathy - a higher F1 score compared to less represented glomerular diseases. For example, tubule-interstitial nephritis (N= 2901) has an F1 score of ± 0.7. For membranous nephropathy (N = 576), the F1 score is ± 0.8.
With respect to table 16 in appendix F, certain statements can be done about

the performance of the classifier.

Classification method

When looking at the classification method, binary classification has higher performance than multilabel classification, which is visualized in figure 24.

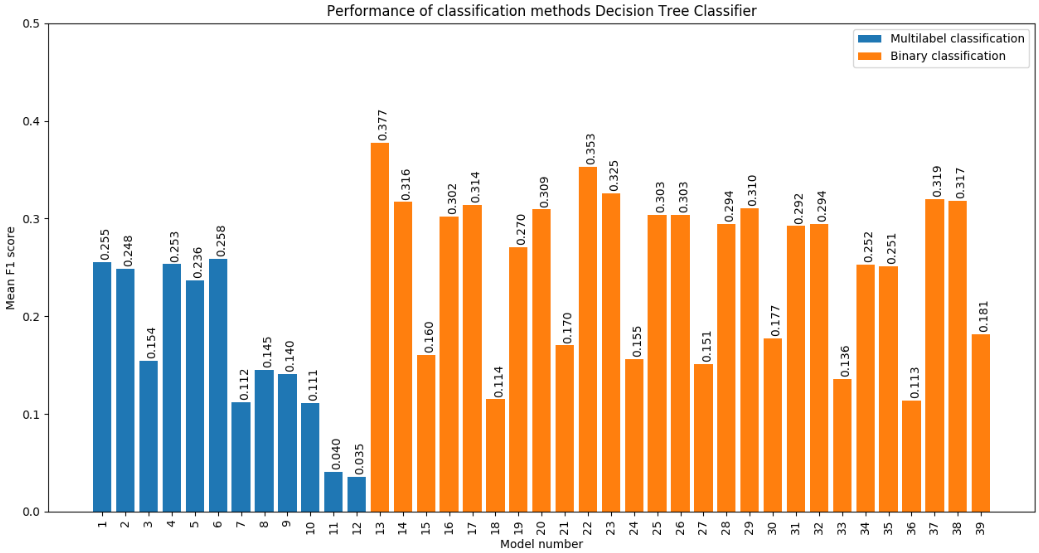


Figure 24: Difference in performance of the models using binary or multilabel classification

The difference in performance can be a result of using separate features for each label with binary classification and thus a more informative input matrix. Overlapping features are more useful for multilabel classification, as some glomerular diseases often present itself secondary to another diseases, as mentioned in section 4.1. However, too much overlapping features lead to no clear distinction between the glomerular diseases, making it harder for a classifier to make correct predictions. We expected the multilabel class to perform better, because of the fact that certain glomerular diseases (such like FSGS) occur secondary. This implies that there are certain relationships between the glomerular diseases, which can be of influence in the process of predicting them.

When using binary classification, there is an average improvement of performance of 27 % with respect to multilabel classification, with no use of performance upgraders or handling class imbalance. This can be seen from the scores of the models 1,2,3 (using multilabel classification) and 13,14,15 (using binary classification) in figure 24. The models can be compared, as all other parameters are equal, except for the use of either multilabel or binary classification. Because there are three different feature subsets, model 1 can be compared with model 13, model 2 with model 14, etc.

When using a random forest classifier and class weights, binary classification (models 34,35,36) has a mean improvement on the mean F1 score with respect to multilabel classification (models 10,11,12). Especially when using random forest feature subset selection, binary classification performs with an improvement 515% better than multilabel classification.

However, when using class weights or random forest classifier, the difference in performance between binary and multilabel classification is nihil.

Feature selection method

When looking at the feature selection method, the filter feature subset selection is of best use in classifying pathology reports, as can be seen in figure 25.

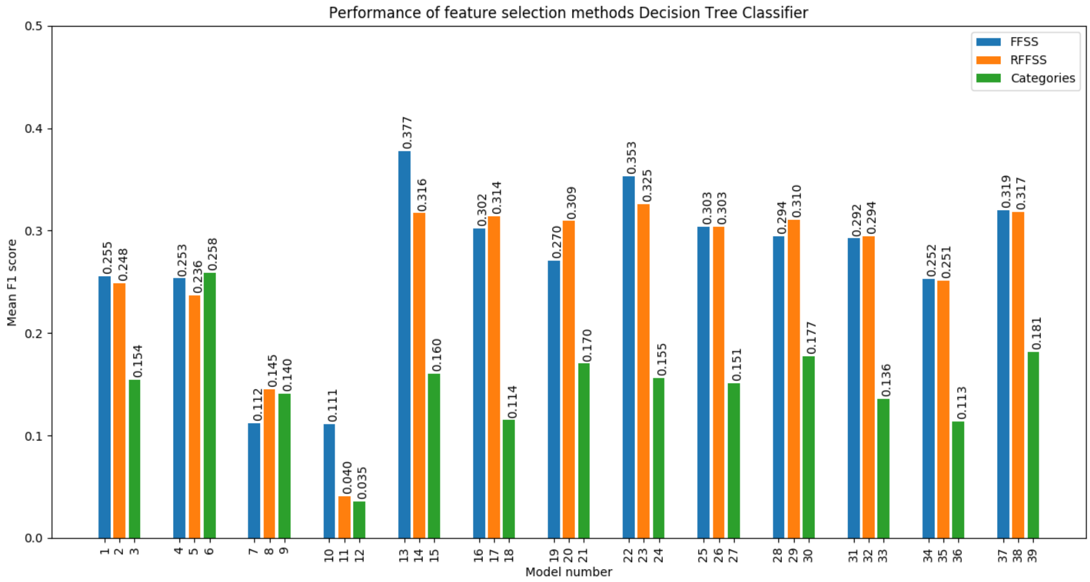


Figure 25: Difference in performance of the models using different feature subsets / types.

The filter feature subset selection (FFSS) performs better than the random forest feature subset selection (RFFSS). A reason for this, is when using RFFSS, the feature types are chosen based on the feature importance. However, these feature types are also based on the frequency of occurrence, causing presumably too much overlap in the feature tokens. This leads to no clear distinction between the glomerular diseases. The categorization method in general does not perform great, which is due to the fact that there are only 25 feature types. Those 25 features are presumably not enough indicators to clearly make distinctions between 32 glomerular diseases. Furthermore, because the biopsy reports do not have clear structure, it is hard to clearly find the pre-defined values of the categorization system.

When using FFSS, there is an average improvement of performance of 13% relative to RFFSS and and improvement of 88% relative to the categorization system. This can be seen in figure 25, where all models using FFSS are compared with all the models using RFFSS and the categorization system.

Performance upgrader

With respect to the performance upgrader, AdaBoost and Random Forest did not have a great influence on the performance of the decision tree classifier. This is not in line with the expectations. Because AdaBoost focuses on rare examples, it was expected that especially with the seen class imbalance, the performance would improve. However, AdaBoost is designed to improve a weak learner, but it needs some base threshold of the base classifier. Because the decision tree itself has an average score below 0.5, it could be possible that there is too much noise and thus results in overfitting of the model. The improvement on using AdaBoost versus not using AdaBoost in figure 26.

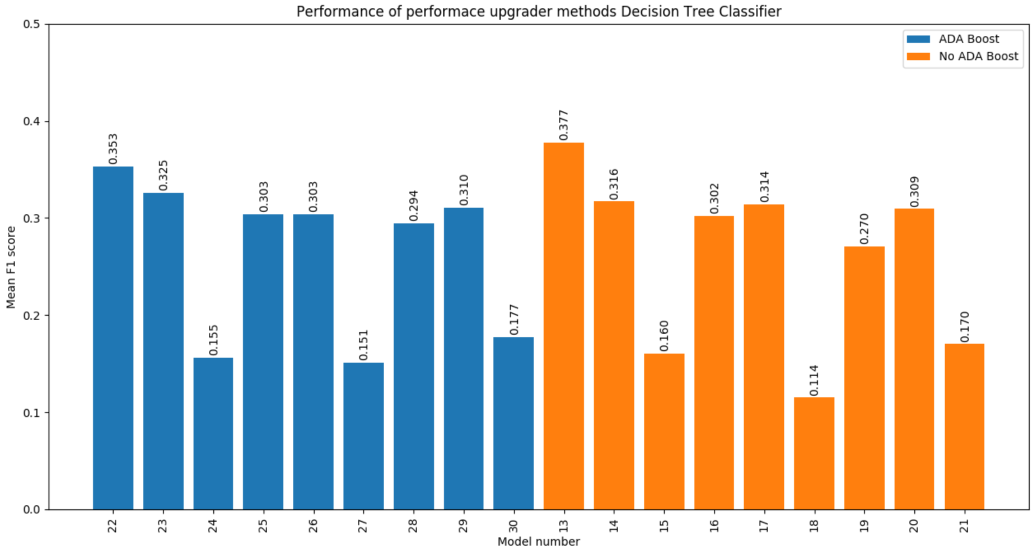


Figure 26: Difference in performance of the models using AdaBoost versus not using AdaBoost.

The nihil influence of the random forest classifier is presumably, because the random forest classifier is based on the mean prediction of the base classifier. This makes the values are not as precise.

Class imbalance

Looking at handling the class imbalance, oversampling has a slightly higher influence on the performance, than class weights using binary classification, which is shown in figure 27. Because oversampling creates more (synthetic) samples, the data becomes more balanced, rather than just giving more weight to a certain class. Because of the high imbalanced data, this could be of influence, but no clear reason for this can be stated.

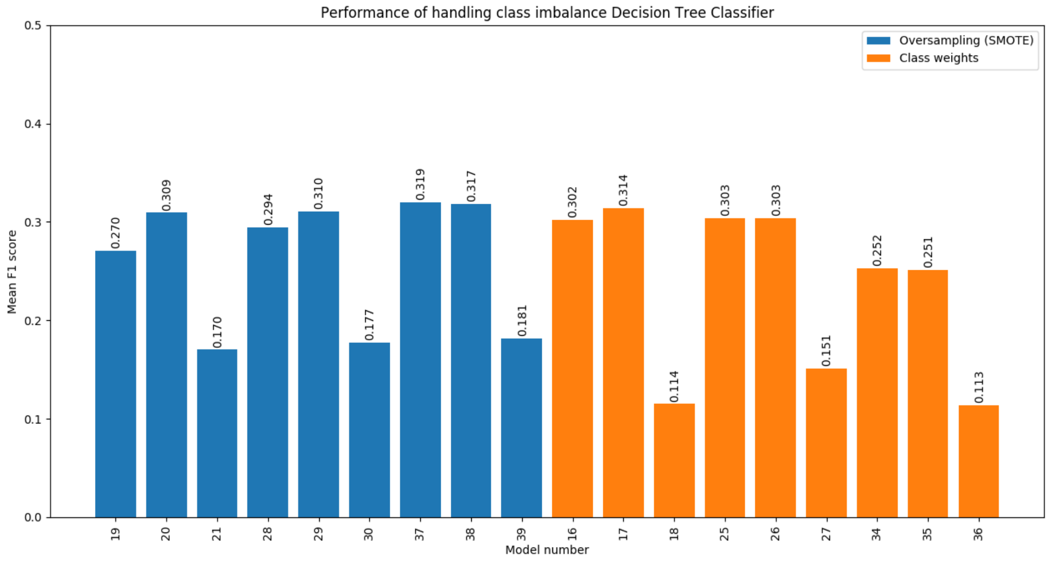


Figure 27: Difference in performance of the models using oversampling or class weights.

4.3.2 Neural network classifier

The neural network classifier has the best performance with the following five models:

1. (a) Model number: #43
   (b) Classification method : Multilabel
   (c) Feature selection : Filter Feature selection
   (d) Performance upgrader : Tensor Flow
   (e) Handling class imbalance : None
2. (a) Model number: #44
   (b) Classification method : Multilabel
   (c) Feature selection : Random Forest Feauture selection
   (d) Performance upgrader : Tensor flow‘
   (e) Handling class imbalance : None
3. (a) Model number: #46
   (b) Classification method : Multilabel
   (c) Feature selection : Filter feature selection
   (d) Performance upgrader : Tensor Flow
   (e) Handling class imbalance : Class Weights
4. (a) Model number: #47
   (b) Classification method : Multilabel

   (c) Feature selection : Random Forest feature selection
   (d) Performance upgrader : Tensor Flow
   (e) Handling class imbalance : Class Weights

5. (a) Model number: #49
   (b) Classification method : Binary
   (c) Feature selection : Filter Feature selection
   (d) Performance upgrader : None
   (e) Handling class imbalance : None

The F1 score of these five models are shown below in table 7.

Disease	**#43**	**#44**	**#46**	**#47**	**#49**
Acute glomerulonephritis	0.1026	0	0.087	0.1154	0.2619
Lupus nephritis	0.7927	0.7857	0.8032	0.749	0.747
Tubule-Interstitial nephritis	0.8034	0.7983	0.8242	0.7814	0.7408
IgA nephropathy	0.7428	0.6929	0.741	0.6704	0.7242
Mcd	0.677	0.613	0.7026	0.6418	0.6549
Fsgs	0.7009	0.5892	0.6921	0.6078	0.6702
Mpgn Type I	0	0	0	0	0
Mpgn Type II	0.5075	0.4719	0.3214	0.5063	0.4286
Mpgn Type III	-	-	-	-	-
Hsp	0.6711	0.6706	0.698	0.6215	0.667
Fibrillary glomerulonephritis	0.6591	0.6038	0.6869	0.7048	0.7312
Amyloidosis AA	0.1951	0.3333	0.02051	0.2857	0.1714
Amyloidosis AL	0.6667	0.6667	0.6928	0.6784	0.6883
Immunotactoid glomerulopathy	0	0	0	0	0
Pauci-immune vasculitis	0.1818	0.1429	0	0	0
Anti-gbm nephritis	0.2791	0.2162	0.1618	0.0719	0.1679
Postinfectious glomerulonephritis	0.1395	0.2308	0	0.16	0.25
Lcdd	0.6239	0.5492	0.625	0.5528	0.6636
Hcdd	-	-	-	-	-
Diabetic nephropathy	0.5168	0.4881	0.6131	0.4869	0.4706
Hereditary nephritis	0.0769	0.3721	0.2	0.1463	0.3429
Lhcdd	0	0	0	0	0
Membranous nephropathy	0.8967	0.8198	0.8932	0.834	0.8755
Mesangiocapillary glomerulonephritis	0.6047	0.5333	0.5797	0.5	0.5278

Disease	**#43**	**#44**	**#46**	**#47**	**#49**
Anca glomerulonephritis	0.7398	0.6659	0.7604	0.6705	0.7322
Acute tubular necrosis	0.2329	0.1463	0.2788	0.2073	0.2595
Cast nephropathy	0.5	0.2667	0.4595	0.4211	0.5952
Tubulopathy	0.4333	0.3663	0.4443	0.3887	0.3291
C3 glomerulopathy	0.2273	0.386	0.1463	0.4138	0.381
Amyloidosis (No type)	0.0816	0.1	0.08	0.2254	0.1481
Focal segmental sclerosing glomerulopathy	0.1299	0.2333	0.1249	0.060	0.2381
Other	0.497	0.4433	0.0472	0.4409	0.3619
Mean	0.427	0.4074	0.4158	0.4071	0.4334

Table 7: Model evaluation neural network classifier; test data = ± 482 samples, training data = ± 4342 samples.

As table 7 shows, the mean F1 score lies around ±0.40. This means that there are less FP's and FN's as with the decision tree classifier. The precision (TP with respect to actual results) and recall (TP with respect to predicted results) are better. Furthermore, just like the decision tree classifier, certain glomerular diseases have a fairly high F1 score.

Below in table 8, the F1 scores of the top five most occurring glomerular diseases (see 23) are shown.

Disease	**#43**	**#44**	**#46**	**#47**	**#49**
Tubule-Interstitial nephritis	0.8034	0.7983	0.8242	0.7814	0.7408
IgA nephropathy	0.7428	0.6929	0.741	0.6704	0.7242
Fsgs	0.7009	0.5892	0.6921	0.6078	0.6702
Membranous nephropathy	0.8967	0.8198	0.8932	0.834	0.8755
Tubulopathy	0.4333	0.3663	0.4443	0.3887	0.3291

Table 8: F1 scores for top 5 most occurring glomerular diseases.

As table 8 shows, the top five most occurring glomerular diseases have a higher F1 score compared to less represented glomerular diseases. For example, tubule-interstitial nephritis (N= 2901), the F1 score is ± 0.8. For membranous nephropathy (N = 576), the F1 score is ± 0.85. This is also due to the relation between the amount of diseases and the F1 score. This is also found in a research of Shah et al., where less data led to less accuracy when using a neural network for classification of clinical notes [68].

With respect to table 16 in appendix F, certain statements can be done about the performance of the classifier.

Classification method

When looking at the classification method, multilabel classification has higher performance than binary classification, which is visualized in figure 28.

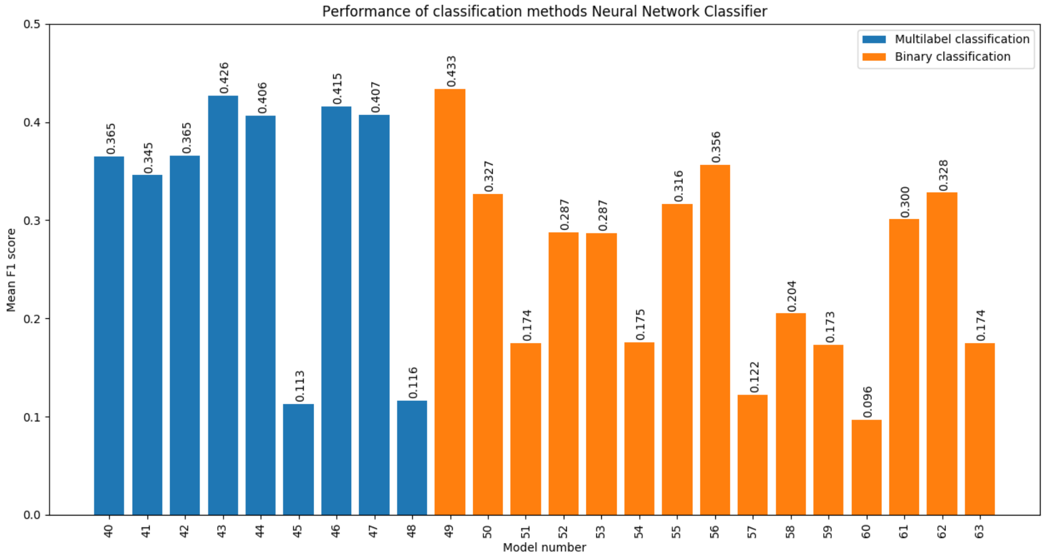


Figure 28: Difference in performance using binary or multilabel classification.

In general, when using multilabel classification, there is an average improvement of 33% with respect to binary classification, with no use of performance upgraders or handling class imbalance. This can be deduced from the scores of the models 40,41,42 and 49,50,51 in figure 28. When using a tensor flow implementation with class weights, multilabel classification (models 46,47,48) has an average improvement of 86% with respect to binary classification (models 58,59,60). This difference in performance between the classification method is in line with the expectations, as certain diseases often present itself secondary with another glomerular disease. Thus there are presumably underlying relations - and thus overlapping features - between the different glomerular diseases.

When looking at the level of the glomerular diseases, FSGS is often seen secondary to IgA nephropathy. When using RFFSS, we see an improvement of performance of 19% for FSGS when using multilabel classification (model 41), with respect to using binary classification (model 50). This is based on comparing models using no performance upgrader or handling class imbalance (see appendix F). For IgA nefropathy, there is an improvement in performance of 52% when using multilabel classification.

When using the categorization system, we see an improvement of performance of 30% for FSGS when using multilabel classification (model 42), with respect to using binary classification (model 51). For IgA nefropathy, there is an improvement in performance of 11% when using multilabel classification.

An interesting result is that for both glomerular diseases, binary classification works best with filter feature selection. This is presumably the result of too little overlapping results, which is a disadvantage when using multilabel classification.

Feature selection method

When looking at the feature selection method, FFSS is of best use in classifying pathology reports, just as with the decision tree classifier, which is shown below in figure 29.

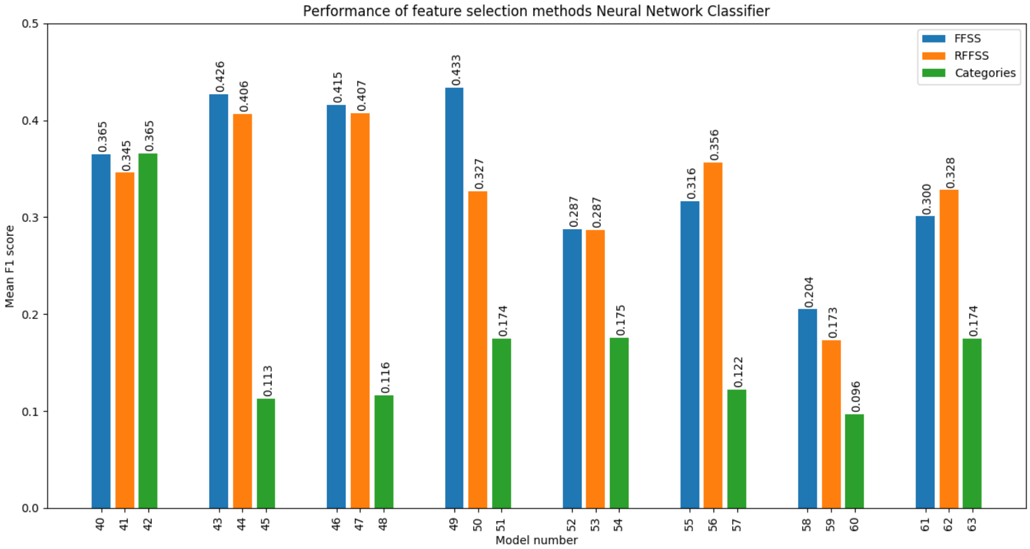


Figure 29: Difference in performance of the models using different feature subsets / types.

When using FFSS, there is an average improve of performance of 6% relatively to RFFSS and an improvement of performance of 136% relatively to the categorization system. This can be seen in figure 29, where all models using FFSS are compared with all the models using RFFSS and the categorization system. Especially compared to the categorization method, FFSS has a better performance. The categorization method in general does not perform great, just as with the decision tree classifier. Also, the same reasoning applies of why FFSS outperforms RFFSS and the categorization system.

Performance upgrader

With respect to the performance upgrader, Tensorflow slightly improves the performance of the neural network classifier, which is shown below in figure 30.

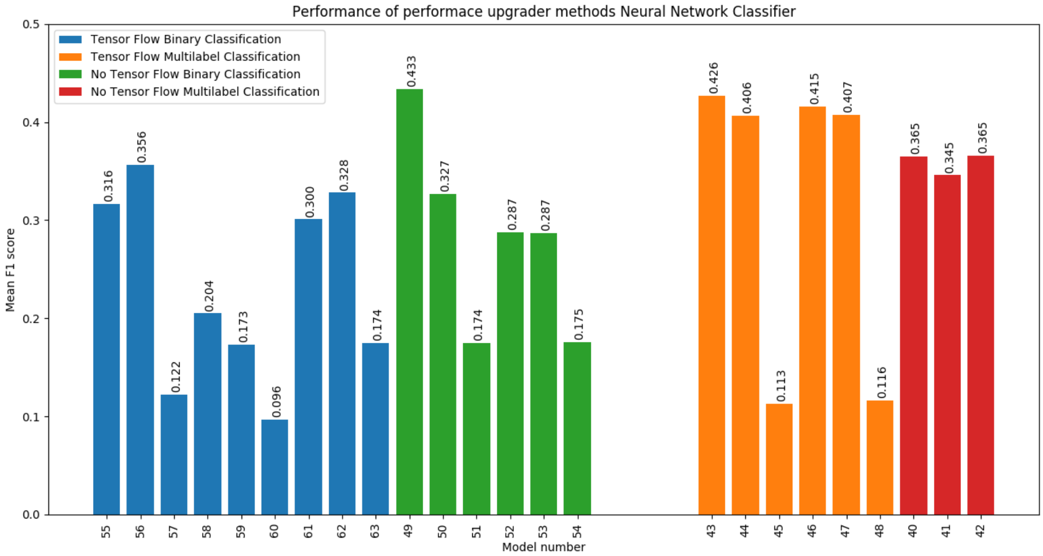


Figure 30: Difference in performance of the models using Tensorflow verus not using Tensorflow.

With the use of Tensorflow, there is an improvement in performance of 17%, using FFSS and multilabel classification (model 43) as opposed to not using Tensorflow (model 40).

Using RFFSS with Tensorfow and multilabel classification (model 44), we see an improvement of 18 % with respect of not using Tensorflow (model 41).

Finally, when using the categorization system, using tensor flow actually decreases the performance of the neural network, which is probably due to overfitting.

When using binary classification, Tensorflow has less impact on the improvement of performance. When using RFFSS with oversampling, the use Tensorflow (model 62) has an improvement in performance of 14% with respect of not using Tensorflow (model 53). However, for the other two feature subsets, the difference is nihil.

All these results are based on the scores of the models shown in figure 30.

Class imbalance

Finally, looking at handling the class imbalance, both using class weights and oversampling does not increase the performance of the classifier. In general, adding class weights or using oversampling to the neural network does not add, but rather slightly decrease the mean performance. This is probably due to the fact that the imbalance can have a positive outcome in classifying glomerular diseases. The occurrence of a glomerular disease can have influence in the performance of the classifier. Over represented glomerular diseases such as tubule-interstitial nephritis are thus predicted correctly by the classifier, which is presented in table 8.

4.4 Subconclusion

The aforementioned results give insights in the answer of subquestion three, mentioned in the introduction.

As table 5 and 7 show, the neural network classifier has better performance in classifying pathology reports, with respect to the decision tree classifier. This is also visualized in figure 31. This is is in line with the research of Menger et al. and Meimandi et al. But as aforementioned, there are also researches where the decision tree classifier performs better than a neural network classifier. In this research, the reason is presumably that a neural network classifier is more robust for unbalanced data and out-sample classification.

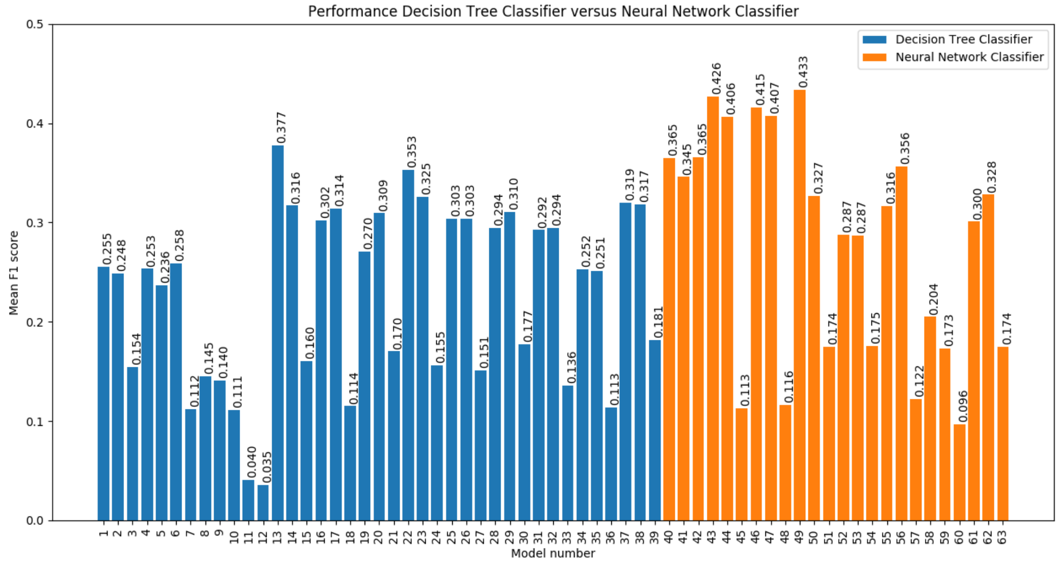


Figure 31: Difference in performance of the models using a decision tree classifier or neural network classifier

With respect to the best performing models of both classifier, the neural network has on average a higher performance of 24% . This can be seen in figure 31, when comparing the scores of models 1-39 and 40-63.

In general, the results in performance of the classify vary per model. For example, binary classification works best for the decision tree classifier, whereas multilabel classification works best for the neural network classifier. It is thus hard to make statements about significant improvement of performance by using certain parameters or methods.

5 Related work

Text mining medical documents, in particular pathology reports is in its infancy. Certain research has been conducted to this phenomenon, but it is mostly based on self-made rules. Certain classifiers are used in order to classify medical documents, or to extract certain relationships of these documents.

In a research of Li et al., machine learning is introduced to state a diagnosis, based on pathology reports of cancer patients [51]. The pathology reports could have multiple categories. The aim of authors was to design a general approach for different prediction categories. The creation of structure data is mostly generated with the help of standard terminologies, such as SNOMED CT. The authors used three classifiers: Naive Bayes, SVM and AdaBoost. Furthermore, they researched the difference of accuracy with experiments, by including or excluding feature selection. Results has shown that there was a high average F1 score (81,3%) for the predominant features, which could be improved by relying on Naive Bayes and feature selection.

Another research of Reihs et al. used a decision tree classifier for automatic classification of histopathological diagnoses. In this research, a dictionary-based decision tree classifier is used. An information extraction module - based on regular expressions - is used in order to extract information about for example tumors dimension, or lymph nodes. This led to certain words that underline a concept for classification. The decision tree itself is build up of nodes, where every node represents a matching word, described by a regular expression pattern, for different spellings an synonyms. Also, every node has a set of processing rules. The results have shown that - based on ICD-10 (International Classification of Diseases 10th revision) - an F1 score of 89,7% is achieved.

Another research of Zhou et al. used a decision tree classifier in order to identify rheumatoid arthritis (RA), based on 9 predictor code groups [83]. The most informative predictors to identify RA are extracted with a decision tree classifier. The decision tree classifier helped with removing codes which cluster with more than one predictor. After this process, patients with RA were classifier. With an overall accuracy of 92,29%, the patients with RA were most of the times classified correctly.

A decision tree classifier is further used in research of Fodeh et al., which compared the classifier with the k-nearest neighbor classifier, the support vector machine and the random forest classifier in order to classify clinical documents with pain assessments [26]. With letting the decision tree classifier grow to its full depth, an overall F1 score of 0.93 is achieved when evaluating the test data with k-fold (with k = 10). However, the random forest classifier scored with an F1 score of 0.94 the best compared to all other used classifiers.

Neural networks are also for classification purposes by different authors.
A research of Shah et al. used neural networks for finding the relation between the symptoms and a disease, based on clinical notes [68]. These relationships are based on word embedding, which are based on neural networks. Furthermore, the authors used Unified Medical Language System (UMLS) to categorize

terms to semantic types. The authors concluded that with the word embedding, the association between the words and the diagnosis were successfully captured. However, they found out that when the instances of training samples are low, the accuracy decreased.

Another research of Kwang et al. also used a artificial neural network for toxicity prediction in radiation oncology. The authors used 10 hidden layers to classify 50 samples. With guidelines regarding toxicity in the bladder and rectum, the authors generated predictor values as input for the neural network classifier. Results have shown that an accuracy of 97,7% for detecting toxicity could be achieved [43].

A research of Rajput et al. used both a single channel and a multi channel deep neural network to detect obesity. With a tensorflow backend, this experimental study focuses also on co-morbidity and has shown promising results. An accuracy of 88,9% of the single channel and 88,06% with the use of the multi channel deep neural network is achieved.

However, there are limited studies which conducted research to diagnose glomerular disorders, based on pathology reports.

6 Conclusion

This thesis describes the classification of biopsy reports of the nephrology department of the Radboud University Medical Center. The biopsy reports describe the findings of histological examination conducted by nephrologists. Each pathology report can have more than one stated diagnosis. By classifying these reports, support can be offered to nephrologists in their decision making process, as this is quite time consuming and complex.

The proposed main question is whether it is possible to design a classification algorithm to predict the diagnosis (or diagnoses) of pathology reports just as good as a nephrologist would do.

A pathology report, existing of natural language, does not have a solid structure. This is of huge influence for a text mining algorithm, especially in the preprocessing face, which is acknowledged by different authors [23] [78]. Feature selection is thus challenging for methods such as filter feature subset selection and random forest feature subset selection, because not enough significant feature types can be produced.

Analyzing medical documents in the healthcare section is often done with classification models, such as decision tree classifiers and neural network classifiers [31] [75]. In this research, we tried to obtain the highest possible performance - expressed by an F1 score - in classifying pathology reports with these two classifiers.

The used method has an advantage over other reviewed classification researches [64] [26], as this research also focused on performance upgrading in a large scale of variability, rather than just focusing on the base classifier.

As a diagnosis (or diagnoses) was not yet linked to a pathology reports, a label extraction process is executed. This resulted in 100% correct extraction. This is the first step in designing classification algorithms and performing model evaluation.

Results have shown that the pathology reports show a huge class imbalance, where some glomerular diseases are with N = 2901 highly over represented, whereas others are with N = 4 rare in occurrence. This is also directly of influence for the performance of both classifiers. Rare glomerular diseases, have a low F1 score of ±0.1, whereas glomerular diseases with higher occurrence have a F1 score of ±0.75.

Overall, the mean F1 score of the top five best models of the decision tree classifier is ± 0.33. For the neural network classifier, this is ± 0.40, which is slightly higher.

The highest mean F1 (± 0.40) and are based on a neural network classifier, using multilabel classification, filter feature subset selection and a tensor flow implementation. For the top five most occurring glomerular diseases, the F1 score is ±0.8 (see table 8). Multilabel classification scores in general higher than binary classification with the use of a neural network classifier. This difference is due to the fact that some glomerular diseases like Focal Segmental Glomerulosclerosis often present itself secondary to other diseases.

Filter feature selection has in general a slightly higher performance than random forest feature subset selection. This is not in line with expectations, as random forest feature subset selection produces more overlapping feature types for the classifier, which could be of advantage in multilabel classification.

In conclusion, it can be stated that the pathology reports are unstructured, making a classification task complicated. In general, the results in performance of the classification varies per model which makes it hard to make certain statements about significant improvement of performance by using certain parameters or methods. When looking at the main question, with an average mean F1 score of ± 0.40, the quality of the classifiers is not enough to fully support nephrologists in their decision making process in classifying pathology reports. For predominant glomerular disease, such as tubular-interstitial nephritis or membranous nefropathy (with an F1 score of ±0.8), the classifiers could be of help in the decision making process. However, for now there is still a human factor needed in order to execute histological examination and state a diagnosis for biopsies. However, with the help of this thesis, a diagnosis (or diagnoses) is/are linked to a pathology report, which can be helpful for further studies, which will be discussed below.

One limitation of this study is the tuning of parameters using only filter feature subset selection. This limitation can have an effect on the performance when using random forest filter subset selection or the categorization system, as the models are more biased using specific feature types.

Another limitation is not using tensor flow to its full potential. In this study, only dense layers are used. However, Keras offers a broad variety of different options to optimize a tensor flow implementation.

Based on the findings in this thesis, several recommendations can be made.
First off, it is recommended to conduct a follow up study, where the categorization system is extended. For this, more guidelines need to be studied. Furthermore, more medical expertise is needed in order to generate a complete categorization system. This brings more general structure to the pathology reports and thus makes it easier to make predictions. After this thesis, I am going to collaborate with the Radboud UMC to further develop this categorization system, extract the pre-defined values out of the pathology reports and study the difference in performance of the current models using the categorization system.

Secondly, it is recommended to generate a structured layout for the pathology reports, as unstructured data forms a huge challenge in the preprocessing phase. An example of a structured layout would be a form, where a nephrologist can choose the pre-defined values of the categories. With the extended category system, this could lead to a good overview of factors on which a glomerular disease is scored. With the help of this new (digital) information, a clear rule-based classification system can be set up in order to predict the glomerular diseases. Furthermore, this categorization system could in the future be of use in combination with image recognition. With an image recognition algorithm, the pre-defined values of the form can be filled out automatically. The rule-based classifier will then directly state a diagnosis. This would be a useful support for nephrologists, as it is a time consuming task to perform histological examination.

Finally it is recommended to conduct a follow up study to the optimization of the tensor flow implementation. In this study, there is no use of word embedding in classifying pathology reports. However, using a conventional network in combination with word embedding, it could be that tensor flow will produce better results. A recent research of Gao et al. used hierarchical self-attention networks in combination with tensor flow, to classify cancer pathology reports [29]. It is recommended to look into this further.

There is no doubt that the importance of data analysis in the health care sector will continue to increase. Using text mining and analysis in order to help predict a diagnosis is one way in which data analysis can make a contribution in the health care sector. A challenge for the future is to make sense of all the data that is available, and this can only be done when IT-professionals and medical specialist work together. Our data science tools will need the data to be structured in a certain way in order to provide a meaningful analysis. There is often not a match between the present data and the data analysis tools we can use to analyse the data yet. I think solving this gap will open up even more possibilities in the future, including more and better diagnosis predictions.

References

[1] Mpgn. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/mpgn/. Accessed on: 21-09-2019.

[2] S.K. Agarwal, S. Sethi, and A.K. Dinda. Basics of kidney biopsy: A nephrologist's perspective. *Indian Journal of Nephrology*, 23(4):243–252, July 2013. doi: 10.4103/0971-4065.114462.

[3] O. Ardhapure, G. Patil, D. Udani, and K. Jetha. Comparative study of classification algorithm for text based categorization. *International Journal of Research in Engineering and Technology*, 5(2), February 2016. doi: 10.15623/ijret.2016.0502037.

[4] Y. Ayar, A. Ersoy, E. Isiktas, G. Ocakoglu, A. Yildiz, A. Oruc, D. Demirayak, I. Bayracki, H. Duger, and T. Bozbudak. The analysis of patients with primary andsecondary glomerular diseases: Asingle-center experience. *Hong Kong Journal of Nephrology*, 19:28–35, October 2016. doi: 10.1016/j.hkjn.2016.05.001.

[5] J.M. Benitez, J.L. Castro, and I. Requena. Are Artificial Neural Networks Black Boxes? *IEEE Transactions on Neural Networks*, 8(5):1156–1164, September 1997. doi: 10.1109/72.623216.

[6] L. Breiman. Random forests. In *Machine Learning*, chapter 8, pages 5–32. Kluwer Academic Publishers, October 2001.

[7] UNC Kidney Center. Al amyloidosis. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/al-amyloidosis/. Accessed on: 21-09-2019.

[8] UNC Kidney Center. Alport syndrome. https://unckidneycenter.org/kidneyhealthlibrary/sglomerular-disease/alport-syndrome/. Accessed on: 21-09-2019.

[9] UNC Kidney Center. Anca vasculitis. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/anca-vasculitis/. Accessed on: 21-09-2019.

[10] UNC Kidney Center. Anti-gbm disease. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/anti-gbm-disease/. Accessed on: 21-09-2019.

[11] UNC Kidney Center. Diabetes. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/diabetes/. Accessed on: 21-09-2019.

[12] UNC Kidney Center. Fibrillary glomerulonephritis (gn). https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/fibrillary-glomerulonephritis-gn/. Accessed on: 21-09-2019.

[13] UNC Kidney Center. Focale segmental glomerulosclerosis (fsgs). https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/focal-segmental-glomerulosclerosis-fsgs/. Accessed on: 18-09-2019.

[14] UNC Kidney Center. Heavy chain deposition disease. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/heavy-chain-deposition-disease/. Accessed on: 21-09-2019.

[15] UNC Kidney Center. Iga nefropathy. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/iga-nephropathy/. Accessed on: 18-09-2019.

[16] UNC Kidney Center. Iga vasculitis (formerly henoch-schönlein purpura or hsp). https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/iga-vasculitis-formerly-henoch-schonlein-purpura-or-hsp/. Accessed on: 18-09-2019.

[17] UNC Kidney Center. Immunotactoid glomerulopathy. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/immunotactoid-glomerulopathy/. Accessed on: 21-09-2019.

[18] UNC Kidney Center. Light chain deposition disease. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/light-chain-deposition-disease/. Accessed on: 21-09-2019.

[19] UNC Kidney Center. Lupus. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/lupus/. Accessed on: 21-09-2019.

[20] UNC Kidney Center. Membranous nefropathy. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/membranous-nephropathy/. Accessed on: 21-09-2019.

[21] UNC Kidney Center. Minimal change disease. https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/minimal-change-disease/. Accessed on: 21-09-2019.

[22] UNC Kidney Center. Post-infectious glomerulonephritis (gn). https://unckidneycenter.org/kidneyhealthlibrary/glomerular-disease/post-infectious-glomerulonephritis-gn/. Accessed on: 21-09-2019.

[23] P. Chatterjee, L.J. Cymberknop, and R.L. Armentano. Nonlinear systems in healthcare towards intelligent disease prediction. In *Nonlinear systems - Volume 2*. 2019.

[24] N.V. Chawla, K.V. Bowyer, L.O. Hall, and W.P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Reserach*, 16:321–357, June 2002. doi: 10.1613/jair.953.

[25] R. Feldman and F. Sanger. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007.

[26] S.J. Fodeh, D. Finch, L. Bouayad, S.L. Luther, H. Ling, R.D. Kerns, and C. Brandt. Classifying clinical notes with pain assessment using machine learning. *Medical Biological Engineering Computing*, 56(7):1285–1292, 2018. doi: 10.1007/s11517-017-1772-1.

[27] A. B. Fogo and M. Kashgarian. *Diagnostic atlas of renal pathology.* Elsevier Limited, 1 edition, 2005.

[28] National Kidney Foundation. Understanding glomerular diseases. https://www.kidney.org/atoz/content/understanding-glomerular-diseases. Accessed on: 18-09-2019.

[29] S. Gao, J.X. Qiu, M. Alawad, J.D. Hinkle, N. Schaefferkoetter, H. Yoon, B. Christian, P.A. Fearn, L. Penberthy, X. Wu, L. Coyle, G. Tourassi, and A. Ramanathan. Classifying cancer pathology reports with hierarchical self-attention networks. *Artificial Intelligence in Medicine*, 101, November 2019. doi: 10.1016/j.artmed.2019.101726.

[30] Google. Introducing tensornetwork, an open source library for efficient tensor calculations. https://ai.googleblog.com/search/label/TensorFlow, June 2019. Accessed on: 23-09-2019.

[31] S. Islam, M. Hasan, X. Wang, H. Germacka, and N.E. Alam. A Systematic Review on Healthcare Analytics: Application and Theoretical Perspective of Data Mining. *Healthcare*, 6(2), 2018. doi: 10.3390/healthcare6020054.

[32] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: with Applications in R.* Springer, 2013.

[33] E. Joyce, P. Glasner, S. Ranganathan, and A. Swiatecka-Urban. Tubulointerstitial nephritis: Diagnosis, treatment and monitoring. *Pediatric Nephrology*, 32(4):577–587, May 2016. doi: 10.1007/s00467-016-3394-5.

[34] H. Kaur and V. Gupta. Indexing process insight and evaluation. 2016 International Conference on Inventive Computation Technologies (ICICT), 2016. doi: 10.1109/INVENTIVE.2016.7830087.

[35] S. Kaushik. https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/, December 2016. Accessed on: 16-10-2019.

[36] Keras. Activations. https://keras.io/activations/. Accessed on: 25-11-2019.

[37] Keras. Keras: The python deep learning library. https://keras.io/. Acessed on: 01-11-2019.

[38] Keras. Losses. https://keras.io/losses/. Accessed on: 25-11-2019.

[39] Keras. Optimizers. https://keras.io/optimizers/. Accessed on: 25-11-2019.

[40] J.J.E. Koopman, Y.K.O. Teng, C.J.F. Boon, L.P. van den Heuvel, T.J. Rabelink, C. van Kooten, and A.P.J. de Vries. Diagnosis and treatment of c3 glomerulopathy in a center of expertise. *The Netherlands Journal of Medicine*, 77(1), January 2019.

[41] K. Kowsari, K.J. Meimandi, M. Heidarysafa, S. Mendu, L.E. Barnes, and D. E. Brown. Text classification algorithms: A survey. *arXiv*, 2019. doi: 10.3390/info10040150.

[42] A. Kumar, V. Dabas, and P. Hooda. Text classification algorithms for mining unstructured data: a swot analysis. *Journal of Biomedical Informetics*, 2018. 10.1007/s41870-017-0072-1.

[43] K. Kwang Hyeon, L. Suk, S. Jang Bo, C. Kyung Hwan, Y. Dae Sik, Y. Won Sup, P. Young Je, K. Chul Yong, and C. Yuan Jie. A text-based data mining and toxicity prediction modeling system for a clinical decision support in radiation oncology: A preliminary study. *Journal of the Korean Physical Society*, 71(4):231–237, August 2017. doi: 10.3938/jkps.71.231.

[44] Scikit Learn. 1.17. neural network models (supervised). https://scikit-learn.org/stable/modules/neural_networks_supervised.html. Accessed on: 01-11-2019.

[45] Scikit Learn. Multiclass and multilabel algorithms. https://scikit-learn.org/stable/modules/multiclass.html. Accessed on: 25-12-2019.

[46] Scikit Learn. sklearn.ensemble.adaboostclassifier. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html. Accessed on: 01-11-2019.

[47] Scikit Learn. sklearn.ensemble.randomforestclassifier. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html. Accessed on: 01-11-2019.

[48] Scikit Learn. sklearn.feature_selection.selectfrommodel. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html. Accessed on: 20-12-2019.

[49] Scikit Learn. sklearn.feature_selection.selectkbest. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html. Accessed on: 01-11-2019.

[50] Scikit Learn. sklearn.tree.decisiontreeclassifier. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html. Accessed on: 01-11-2019.

[51] Y. Li and D. Martinez. Information extraction of multiple categories from pathology reports. In *ALTA*, 2010.

[52] C.D. Manning, Raghavan. P., and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[53] K.D. McClatchey. *Clinical Laboratory Medicin*. Lippincott Williams Wilkins, 2012.

[54] V. Menger, F. Scheepers, and M. Spruit. Comparing deep learning and classical machine learning approaches for predicting inpatient violence incidents from clinical text. *Applied Sciences*, 8(6), 2018. doi: 10.3390/app8060981.

[55] M. Mubaral and H. Nasri. Significance of segmental glomerulosclerosis in iga nephropathy: What is the evidence? *Journal of Renal Injury Prevention*, 2(4):131–115, 2013. doi: 10.12861/jrip.2013.36.

[56] S. *Towards Data Science* Narkhede. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62, 2018. Accessed on: 03-11-2019.

[57] Nierstichting. Feiten en cijfers. https://www.nierstichting.nl/over-nieren/hoe-werken-je-nieren/feiten-en-cijfers/. Accessed on: 18-09-2019.

[58] T. Nijenhuis and J.F.M. Wetzels. Nierziekten. In *Leerboek Interne Geneeskunde*, chapter 14, pages 411–460. Bohn Stafleu van Loghum, 2004.

[59] National Institute of Diabetes, Digestive, and Kidney Diseases. Glomerular diseases. https://www.niddk.nih.gov/health-information/kidney-disease/glomerular-diseases. Accessed on: 18-09-2019.

[60] National Institute of Diabetes, Digestive, and Kidney Diseases. Glomerular diseases. https://www.niddk.nih.gov/health-information/kidney-disease/glomerular-diseases. Accessed on: 23-09-2019.

[61] University of Rochester Medical Center. Albumin (blood). https://www.urmc.rochester.edu/encyclopedia/content.aspx?contenttypeid=167&contentid=albumin_blood. Accessed on: 23-09-2019.

[62] OnzeTaal. Woordfrequentie. https://onzetaal.nl/taaladvies/woordfrequentie, June 2019. Accessed on: 04-10-2019.

[63] Harvard Health Publishing. Edema. https://www.health.harvard.edu/a_to_z/edema-a-to-z. Accessed on: 15-12-2019.

[64] K. Rajput, G. Chetty, and R. Davey. Obesity and co-morbidity detection in clinical text using deep learning and machine learning techniques. In *Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*. IEEE, December 2018. doi: 10.1109/APWConCSE.2018.00017.

[65] S. Russel and P. Norivg. *Artificial Intelligence: A Modern Approach*. 3 edition.

[66] E. Schapire. The boosting approach to machine learning: An overview. In D.D. Denisopn, M.H. Hansen, C.C. Holmes, B. Mallick, and B. Yu, editors, *Nonlinear Estimation and Classification*, chapter 8, pages 149–171. Springer, 2003.

[67] S. Sethi, C.M. Nester, and R.J.H Smith. Membranoproliferative glomerulonephritis and c3 glomerulopathy: Resolving the confusion. *Kidney International*, 81(5):434–441, March 2012. doi: 10.1038/ki.2011.399.

[68] S. Shah, X. Luo, S. Kanakasabai, R. Tuason, and G. Klopper. Neural networks for mining the associations between diseases and symptoms in clinical notes. *Health Information Science and Systems*, 7(1), December 2019. doi: 10.1007/s13755-018-0062-0.

[69] B. Shen. *Bioinformatics for Diagnosis, Prognosis and Treatment of Complex Diseases.* Springer Science Business Media, January 2013. doi: 10.1007/978-94-007-7975-4.

[70] Federatie Medisch Specialisten. Bepaling van nierfunctie en albuminurie. Accessed on: 20-09-2019.

[71] Federatie Medisch Specialisten. Diagnostiek bij hematurie: urineonderzoek. https://richtlijnendatabase.nl/richtlijn/hematurie/diagnostiek_hematurie_-_urineonderzoek.html. Accessed on: 20-09-2019.

[72] Federatie Medisch Specialisten. Stadiëring bij chronische nierschade. https://richtlijnendatabase.nl/richtlijn/chronische_nierschade_cns/diagnostiek_en_stadiering_bij_cns/stadiering_bij_chronische_nierschade.html. Accessed on: 20-09-2019.

[73] J Taeho. *Text Mining : Concepts, Implementation, and Big Data Challenge*, volume 45 of *Studies in Big Data*. Springer International Publishing, 2019. doi: 10.1007/978-3-319-91815-0.

[74] P. Tan, M. Steinbach, and V. Kumar. *Introduction to data mining*. Pearson Education, Inc., 2006.

[75] V. Tang, P.K.Y. Siu, K.L. Choy, H.Y. Lam, G. T.S.M. Ho, C.K.M. Lee, and Y.P. Tsang. An adaptive clinical decision support system for serving the elderly with chronic diseases in healthcare industry. *Expert Systems*, 36(2), April 2019. doi: 10.1111/exsy.12369.

[76] J. Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcome. *Journal of Clinical Epidemiology*, 49.

[77] Radboud UMC. Department nephrology. https://www.radboudumc.nl/en/research/departments/nephrology. Accessed on: 18-09-2019.

[78] Y. Wang, L. Wang, M. Rastegar-Mojarad, S. Moon, F. Shen, N. Afzal, S. Liu, Y. Zeng, S. Mehrabi, S. Sohn, and H. Liu. Clinical information extraction applications: A literature review. *Journal of Biomedical Informatics*, 77:34–49, January 2018. doi: 10.1016/j.jbi.2017.11.011.

[79] C.L. Weber, C.L. Rose, and A.B. Magil. Focal segmental glomerulosclerosis in mild iga nephropathy: a clinical-pathologic study. *Nephrology Dialysis Transplantation*, 24(2):483–488, 2009. doi: 10.1093/ndt/gfn513.

[80] J Wetzels. Diagnosis of glomerular diseases. RUMC.

[81] E.M. Witteman, H.L.A. Janssen, B.P.C. Hazenberg, and S. Janssen. Amyloidosis; pathogenese en therapie. *Nederlands Tijdschrift voor Geneeskunde*, pages 2318–2322, 1992.

[82] WolframMathWorld. Tensor contraction. http://mathworld.wolfram.com/TensorContraction.html. Accessed on: 23-09-2019.

[83] S. Zhou, F. Fernandez-Guitierrez, J. Kennedy, R. Cooksey, M. Atkinson, S. Denaxas, S. Siebert, W.G. Dixon, T.W. O'Neill, E. Choy, C. Sudlow, and S. Brophy. Defining disease phenotypes in primary care electronic health records by a machine learning approach: A case study in identifying rheumatoid arthritis. *PLOS ONE*, 5 2016. doi: 10.1371/journal.pone.0154515.

Appendices

A Glomerular diseases

This appendix shows the most common glomerular diseases, with their explanation in table 9.

Diseases	Explanation
Membranous Nephropathy (MN)	Also called Membranous Glomerulopathy. Filters of the kidneys are affected, in which proteins decrease [20]. Characterised by global subepithelial deposits. Diagnosed with light microscopy, immunofluorescence and electron microscopy. [27]
Minimal Change Disease (MCD)	Leaking of proteins in the filters of the kidneys [21]. Characterised by structurally normal glomeruli. May also occur in older adults who have nonspecific focal areas of tubulointerstitial scarring. Diagnosed with light microscopy [27].
Focal Segmental Glomeruloscleroses (FSGS)	Developing scar tissue on the filter of the kidneys [13]. Characterised by sclerosis of a portion of the glomeruli. Diagnosed with light microscopy [27].
IgA Nephropathy (IgAN)	Damage of the filters of the kidneys, caused by Immunoglobuline A (IgA) that is stuck in the filters [15]. Characterised by hematuria and varying proteinuria. Diagnosed with light microscopy, immunofluorescence and electron microscopy. [27].

Henoch Schonlein Purpura (HSP)	Inflammation of blood vessels, causing rash on the skin. Mostly seen with (?) children [16]. Characterised by purpuric lesions of the skin, arthiritis, and gastrointestinal hemorrhage. Diagnosed with light microscopy, immunofluorescence and electron microscopy. [27].
Membranoproliferative Glomerulonephritis (MPGN) Type I	Problems with the immune system, where the glomeruli are damaged [1]. Characterised by hypertension and impaired renal function [27]. Diagnosed with light microscopy, immunofluorescence and electron microscopy.
Membranoproliferative Glomerulonephritis (MPGN) Type II	Characterised by decrease in complements (in particular C3) and hypertension [27]. [1].
Membranoproliferative Glomerulonephritis (MPGN) Type III	*See MPGN Type I.*
C3 Glomerulopathy	Rare renal disease. Characterised by deposition of complement factor C3, a protein involved in the immune system [40]. Different from MPGN Type II [67].
Dense Deposit Disease	Very rare renal disease. *See MPGN Type II*
Fibrillary Glomerulonephritis (GN)	High production of proteins, which causes swelling of the glomeruli [12]. Characterised by hematuria [27].

Amyloidosis AA	Blood illness, in which the amyloid protein type AA is build up [81]. They can get stuck in the glomeruli filters, causing leakage of protein in the urine [7].
Amyloidosis AL	The build of amyloid protein type AL. They can get stuck in the glomeruli filters, causing leakage of protein in the urine [7]
Immunotactoid Glomerulopathy	Abnormal deposition of antibodies in the kidneys. [17]. Characterised by parallel arrays (tactoids) [27].
Lupus Nephritis	Forming of anitbodies against itself, which will attack the kidneys. [19]. There are six classes indicating the severity of Lupus [27].
Pauci-immune Vasculitis	Inflammation of the blood vessels, because antibodies attack white blood cells. Associated with ANCA [9].
Anti-Glomerular Basement Membrane (GBM) Nephritis	Targeting of antibodies towards the capillary blood vessels of the kidney [10]. Characterised by damage to the GMB [27].
Postinfectious Glomerulonephritis (GN)	Infection of the kidneys, causing swollen glomeruli filters [22]
Light Chain Deposition Disease (LCDD)	Antibodies exists of protein segments: light chains and heavy chains. With LCDD, too many light chains are made, causing them to get stuck in the kidneys. [18].

Heavy-Chain Deposition Disease (HCDD)	Deposition of heavy chains in the kidneys [14].
Tubule-Interstitial Nephritis	Inflammation in the renal tubules, causing injury [33].
Diabetic Nephropathy Diabetic Glomerulosclerosis	Scarring of the glomeruli, due to diabetes [11]. Characterised by the progression of microalbuminuria to proteinuria [27].
Hereditary Nephritis (M. Alport)	Disease of the glomeruli filters, due to gene mutations. [8]. Characterised by hematuria [27].

Table 9: Most common glomerular diseases [80]

B Python libraries

This appendix shows the (sub)libraries that are used for designing the text algorithm in table 10.

Library	Sublibrary
Numpy	genfromtxt
Sklearn	SelectKBest
	chi2
	SelectFromModel
	tree
	AdaBoostClassifier
	MLPClassifier
	StratifiedKfold
	confusion_matrix
	RandomForestClassifier
	multilabel_confusion_matrix
Skmultilearn	iterative_train_test_split
	IterativeStratification
Keras	Sequential
	Dense
Tensorflow	-
Random	Randrange
imblearn	SMOTE
Pandas	-
Re	-
Collections	Counter
Matplotlib.pyplot	-
Itertools	-
String	-
Argparse	-
Nltk	word_tokenize
Dominate	document

Table 10: Used Python (sub)libraries to design the text algorithm.

C Data preprocessing

This appendix shows the data preprocessing steps. The list of the removed frqequent dutch words, the glomerular diseases classification system and the metadata with respect to random forest feature selection are described below.

C.1 Removed dutch words

The removed frequent dutch words in the pathology reports are:

- de
- het
- een
- dat
- je
- in
- maar
- is
- te
- met
- die
- worden
- op
- en
- voor
- van
- dit
- zijn
- of

C.2 Glomerular diseases classification system

The glomerular diseases classification system is shown in table 11.

GD_id	**Category**	**Value**
GDid_01	name	acute glomerulonefritis
	synonym	acute glomerulonephritis
	synonym	immuuncomplex glomerulonefritis
GDid_02	name	lupus nephritis
	synonym	lupusnefritis
	synonym	lupus nefritis
	synonym	lse nefritis
	synonym	lse nephritis
	synonym	lupus
GDid_03	name	interstitiele nefritis/ interstitiele fibrose tubulaire atrofie
	synonym	interstitiele nefritis
	synonym	interstitiele fibrose tubulaire atrofie
	synonym	tubulaire atrofie
	synonym	tubulointerstitiele nefritis
	synonym	tubulo-interstitiele nefritis
	synonym	tubulo interstitiele nefritis
	synonym	interstitiele fibrose
	synonym	tubulo interstitiele ontstekings
	synonym	tubulo interstitiele
	synonym	interstitiele ontsteking
	synonym	interstitiele ontstekingscomponent
	synonym	interstiitele fibrose

GD_id	Category	Value
GDid_04	name	iga nefropathie
	synonym	iga nephropathie
	synonym	iga glomerulonefritis
	synonym	iga glomerulonephritis
GDid_05	name	mcd
	synonym	minimal change
	synonym	minimal change disease
GDid_06	name	fsgs
	synonym	focale glomerulosclerose
	synonym	focale glomeruloscleroses
	synonym	focale glomerulosclerose
	synonym	focale segmentale glomerulosclerose
GDid_07	name	mpgn tpye i
	synonym	mpgn 1
	synonym	mpgn1
GDid_08	name	mpgn type ii
	synonym	mpgn2
	synonym	mpgn 2
	synonym	ddd
	synonym	dense deposit disease
GDid_09	name	mpgn tpye iii
	synonym	mpgn 3
	synonym	mpgn3

GD_id	Category	Value
GDid_10	name	hsp
	synonym	henoch-schonlein purpura
	synonym	henoch schonlein purpura
	synonym	henoch-schonlein
	synonym	henoch schonlein
GDid_11	name	fibrillaire glomerulonefritis
	synonym	fibrillaire glomerulonephritis
	synonym	fibrillaire glomerulopathie
GDid_12	name	amyloidosis aa
	synonym	aa amyloidosis
	synonym	amyloide aa
	synonym	amyloidose aa
	synonym	aa amyloid
	synonym	amyloid aa
	synonym	aa amyloidose
	synonym	aa type
	synonym	type aa
GDid_13	name	amyloidosis al
	synonym	al amyloid
	synonym	al amyloidose
	synonym	al amyloidosis
	synonym	amyloidose al
	synonym	amyloid al
	synonym	type al
	synonym	al type

GD_id	**Category**	**Value**
GDid_14	name	immunotactoid glomerulopathie
	synonym	immunotactoide glomerulopathie
GDid_15	name	crescentische glomerulonefritis
	synonym	pauci-immune crescentische glomerulonefritis
	synonym	pauci immune crescentische glomerulonefritis
	synonym	extracapillair glomerulonefritis
	synonym	pauci-immune extracapillair glomerulonefritis
	synonym	pauci immune extracapillair glomerulonefritis
	synonym	pauci-immune anca glomerulonefritis
	synonym	pauci immune extracapillair
	synonym	pauci-immune extracapillaire
	synonym	pauci-immune extracapillair
	synonym	pauci-immune extracapillaire glomerulonefritis
	synonym	pauci immune extracapillaire glomerulonefritis
	synonym	extracapillaire glomerulonefritis
	synonym	pauci-immune glomerulonefritis
	synonym	pauci-immune necrotiserende glomerulonefritis
	synonym	pauci immune necrotiserende glomerulonefritis
	synonym	pauci immune glomerulonefritis
	synonym	pauci-immuun vasculitis
	synonym	pauci-immuun patroon
	synonym	pauci immuun patroon
	synonym	crescentische glomerulaire afwijkingen

GD_id	Category	Value
GDid_16	name	anti-gbm nefritis
	synonym	anti-gbm nephritis
	synonym	anti gbm nefritis
	synonym	anti gbm glomerulonefritis
	synonym	anti-gbm glomerulonefritis
GDid_17	name	postinfectieuze glomerulonephritis
	synonym	postinfectieuze glomerulonefritis
	synonym	post-infectieuze glomerulonefritis
	synonym	post infectieuze glomerulonefritis
	synonym	glomerulonefritis postinfectieus
	synonym	glomerulonephritis postinfectieus
GDid_18	name	lhcdd
	synonym	light en heavy deposition disease
	synonym	light and heavy chain depositie ziekte
GDid_19	name	lcdd
	synonym	light chain disposition disease
	synonym	light chain
	synonym	light chain depositie
	synonym	light chain depositie ziekte
	synonym	lcd
	synonym	lichte keten cast
	synonym	lichte keten
GDid_20	name	hcdd
	synonym	heavy chain disposition disease
	synonym	heavy chain depositie
	synonym	heavy chain depositie ziekte
	synonym	hcd

GD_id	**Category**	**Value**
GDid_21	name	diabetische nefropathie
	synonym	diabetische glomeruloscleroses
	synonym	diabetische glomerulosclerose
	synonym	diabetische glomerulopathie
GDid_22	name	hereditaire nefritis
	synonym	m alport
	synonym	morbus alport
	synonym	alport
	synonym	hereditaire nephritis
GDid_23	name	membraneuze nefropathie
	synonym	membraneuze glomerulonefritis
	synonym	membraneuze glomerulopathie
GDid_24	name	mesangiocapillaire glomerulonefritis
	synonym	membranoproliferatieve glomerulonefritis
GDid_25	name	anca glomerulonefritis
	synonym	anca glomerulonephritis
	synonym	anca ziekte
	synonym	anca
GDid_26	name	acute tubulusnecrose
GDid_27	name	cast nefropathie
	synonym	cast nephropathie
	synonym	cast-nefropathie

GD_id	**Category**	**Value**
GDid_28	name	tubulopathie
GDid_29	name	c3 nefropathie
	synonym	c3 nephropathie
GDid_30	name	amyloidosis (geen type)
	synonym	amyloidosis
	synonym	amyloidose
	synonym	amyloid
Gdid_31	name	focale segmentale scleroserende glomerulonefritis

Table 11: Glomerular Disease Classification System; A unique *GDid* for each glomerular disease and their corresponding names and synonyms.

C.3 Metadata random forest feature selection

The amount of features for each glomerular disease, based on random forest feature selection is shown in table 12.

Diseases	**Amount**
Acute Glomerulonephritis	45
Lupus Nephritis	20
Tubule-Interstitial nephritis	39
Iga nephropathy	42
Mcd	26
Fsgs	42
MPGN Type I	50
MPGN Type II	30
MPGN Type III	0
Hsp	32
Fibrillary glomerulonephritis	30
Amyloidosis AA	22
Amyloidosis AL	14
Immunotactoid Glomerulopathy	58
Pauci-immune vasculitis	39
Anti-gbm nephritis	30
Postinfectious glomerulonephritis	56

Lcdd	27
Hcdd	0
Diabetic nephropathy	42
Hereditary nephritis	35
Lhcdd	54
Membranous nephropathy	15
Mesangiocapillary glomerulonephritis	28
Anca glomerulonephritis	I24
Acute tubular necrosis	47
Cast nephropathy	26
Tubulopathy	38
C3 nephropathy	37
Amyloidosis (No type)	39
Focal segmental sclerosing glomerulopathy	55
Other	35

Table 12: Amount features for each diseases for Random Forest classifier.

D Categorization system

This appendix shows the categorization system used for feature selection, in table 13.

Category	Value
Endocapillary proliferation	None
	Minimal
	Present
	Unknown
Extracapillary proliferation	None
	Minimal
	Present
	Unknown
Mesangial proliferation	None
	Minimal
	Present
	Unknown
Mesangiocapillary proliferation	None
	Minimal
	Present
	Unknown
Foot effacement	None
	Minimal
	Partial
	Complete
	Unknown
Hyalinosis	Present
	Not present
	Present

Category	**Value**
Hypertrophy	Present
	Not present
	Unknown
Adhesion	Present
	Not present
	Unknown
Glomerular hematuria	Present
	Not present
	Unknown
Proteinuria	Present
	Not present
	Unknown
Acute kidney insufficiency	Present
	Not present
	Unknown
Trombosis	Present
	Not present
	Unknown
Focal segmental sclerosis	Present
	Not present
	Unknown
C3 deposition	Positive
	Negative
	Unknown
C1q deposition	Positive
	Negative
	Unknown
C4 deposition	Positive
	Negative
	Unknown

Category	**Value**
aGBM level	Positive
	Negative
	Unknown
ANCA level	Positive
	Negative
	Unknown
IgA level	0+
	1+
	2+
	3+
	4+
	Trace
	Unknown
IgM level	0+
	1+
	2+
	3+
	4+
	Trace
	Unknown
IgG level	0+
	1+
	2+
	3+
	4+
	Trace
	Unknown

Category	**Value**
Kappa level	0+
	1+
	2+
	3+
	4+
	Trace
	Unknown
Lambda level	0+
	1+
	2+
	3+
	4+
	Trace
	Unknown
Amount Sclerosis	0-25%
	25-50%
	50-75%
	75-100%
	Unknown
Diabetes type	Type I
	Type II
	No type
	Unknown

Table 13: Categorization system with pre-defined values.

E Model Building

This appendix shows the specific parameters for two classifiers: decision tree classifier (see table 14) and neural network classifier (see table 15).

#	Multi-label classification	Binary classification	Filter feature selection	RF feature selection	Categories	ADA Boost	Tensor Flow	RF	Class weights	Oversampling
1	X	-	X	-	-	-	-	-	-	-
2	X	-	-	X	-	-	-	-	-	-
3	X	-	-	-	X	-	-	-	-	-
4	X	-	X	-	-	-	-	-	X	-
5	X	-	-	X	-	-	-	-	X	-
6	X	-	-	-	X	-	-	-	X	-
7	X	-	X	-	-	-	-	X	-	-
8	X	-	-	X	-	-	-	X	-	-
9	X	-	-	-	X	-	-	X	-	-
10	X	-	X	-	-	-	-	X	X	-
11	X	-	-	X	-	-	-	X	X	-
12	X	-	-	-	X	-	-	X	X	-
13	-	X	X	-	-	-	-	-	-	-
14	-	X	-	X	-	-	-	-	-	-
15	-	X	-	-	X	-	-	-	-	-
16	-	X	X	-	-	-	-	-	X	-
17	-	X	-	X	-	-	-	-	X	-
18	-	X	-	-	X	-	-	-	X	-
19	-	X	X	-	-	-	-	-	-	X
20	-	X	-	X	-	-	-	-	-	X
21	-	X	-	-	X	-	-	-	-	X

22	-	X	X	-	-	X	-	-	-	-
23	-	X	-	X	-	X	-	-	-	-
24	-	X	-	-	X	X	-	-	-	-
25	-	X	X	-	-	X	-	-	X	-
26	-	X	-	X	-	X	-	-	X	-
27	-	X	-	-	X	X	-	-	X	-
28	-	X	X	-	-	X	-	-	-	X
29	-	X	-	X	-	X	-	-	-	X
30	-	X	-	-	X	X	-	-	-	X
31	-	X	X	-	-	-	-	X	-	-
32	-	X	-	X	-	-	-	X	-	-
33	-	X	-	-	X	-	-	X	-	-
34	-	X	X	-	-	-	-	X	X	-
35	-	X	-	X	-	-	-	X	X	-
36	-	X	-	-	X	-	-	X	X	-
37	-	X	X	-	-	-	-	X	-	X
38	-	X	-	X	-	-	-	X	-	X
39	-	X	-	-	X	-	-	X	-	X

Table 14: Models Decision Tree Classifier; 'X' when parameter is "on", '-' when parameter is "off".

#	Multi-label classification	Binary classification	Filter feature selection	RF feature selection	Categories	ADA Boost	Tensor Flow	RF	Class weights	Oversampling
40	X	-	X	-	-	-	-	-	-	-
41	X	-	-	X	-	-	-	-	-	-
42	X	-	-	-	X	-	-	-	-	-
43	X	-	X	-	-	-	X	-	-	-
44	X	-	-	X	-	-	X	-	-	-
45	X	-	-	-	X	-	X	-	-	-
46	-	X	X	-	-	-	X	-	X	-
47	-	X	-	X	-	-	X	-	X	-
48	-	X	-	-	X	-	X	-	X	-
49	-	X	X	-	-	-	-	-	-	-
50	-	X	-	X	-	-	-	-	-	-
51	-	X	-	-	X	-	-	-	-	-
52	-	X	X	-	-	-	-	-	-	X
53	-	X	-	X	-	-	-	-	-	X
54	-	X	-	-	X	-	-	-	-	X
55	-	X	X	-	-	-	X	-	-	-
56	-	X	-	X	-	-	X	-	-	-
57	-	X	-	-	X	-	X	-	-	-
58	-	X	X	-	-	-	X	-	X	-
69	-	X	-	X	-	-	X	-	X	-
60	-	X	-	-	X	-	X	-	X	-
61	-	X	X	-	-	-	X	-	-	X
62	-	X	-	X	-	-	X	-	-	X
63	-	X	-	-	X	-	X	-	-	X

Table 15: Models Neural Network Classifier; ’X’ when parameter is “on”, ’-’ when parameter is “off”.

F Model evaluation

This appendix shows the k-fold model evaluation of the 63 models. The F1 score of the 63 models of each glomerular disease, including the mean score are shown in tabel 16.

Disease	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**	**#8**	**#9**	**#10**
Acute glomerulonephritis	0	0.0423	0.0194	0.0339	0.0441	0.0508	0	0	0	0
Lupus nephritis	0.6228	0.6208	0.3666	0.631	0.5817	0.6208	0.3322	0.5722	0.5521	0.38
Tubule-Interstitial nephritis	0.7302	0.717	0.6741	0.7468	0.716	0.7265	0.8457	0.8354	0.7793	0.8417
IgA nephropathy	0.5408	0.5353	0.5328	0.5166	0.5027	0.4795	0.1488	0.2969	0.3397	0.1139
Mcd	0.5271	0.4639	0.0496	0.4956	0.4062	0.4957	0.2849	0.4496	0.4138	0.3059
Fsgs	0.5208	0.4799	0.4343	0.5028	0.4632	0.482	0.3635	0.4163	0.4173	0.3282
Mpgn Type I	0		0	0	0	0	0	0	0	0
Mpgn Type II	0.1493	0.1579	0.0541	0.1404	0.1587	0.1176	0	0	0	0
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0.4267	0.3864	0.1084	0.3742	0.3721	0.4551	0	0	0	0
Fibrillary glomerulonephritis	0.1702	0.2268	0.1633	0.2136	0.22	0.2963	0	0	0	0
Amyloidosis AA	0.1333	0.0833	0	0	0.0727	0.0345	0	0	0	0
Amyloidosis AL	0.3602	0.2238	0.1805	0.4146	0.3694	0.895	0	0	0	0
Immunotactoid glomerulopathy	0	0	0	0	0	0	0	0	0	0
Pauci-immune vasculitis	0.0432	0.0683	0.0385	0.0628	0.0841	0.11	0	0	0	0
Anti-gbm nephritis	0	0.1404	0.0526	0	0.1017	0.0784	0	0	0	0
Postinfectious glomerulonephritis	0.0634	0.0896	0.0294	0.1071	0	0.03	0	0	0	0

Disease	**#1**	**#2**	**#3**	**#4**	**#5**	**#6**	**#7**	**#8**	**#9**	**#10**
Lcdd	0.3362	0.3734	0.25	0.2583	0.288	0.2553	0	0.1353	0.0923	0
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0.2715	0.1386	0.2688	0.2483	0.1695	0.1567	0	0	0	0
Hereditary nephritis	0.0588	0.1277	0	0	0.0417	0.0392	0	0	0	0
Lhcdd	0	0	0	0	0	0	0	0	0	0
Membranous nephropathy	0.7661	0.7432	0.4398	0.7983	0.7519	0.7714	0.7346	0.7889	0.7864	0.7259
Mesangiocapillary glomerulonephritis	0.1508	0.1883	0.1667	0.1892	0.2183	0.1667	0	0	0	0
Anca glomerulonephritis	0.52	0.4742	0.3487	0.5149	0.4942	0.4576	0.271	0.4214	0.409	0.2447
Acute tubular necrosis	0.1031	0.0566	0	0.0778	0.0793	0.0833	0	0	0	0
Cast nephropathy	0.3434	0.2752	0.0769	0.3011	0.1228	0.145	0	0	0	0
Tubulopathy	0.2652	0.278	0.1527	0.2599	0.2564	0.268	0	0.0376	0.029	0
C3 glomerulopathy	0.125	0.0727	0	0.1786	0.0727	0.0392	0	0	0	0
Amyloidosis (No type)	0	0.1096	0.0339	0.0444	0.0769	0.0833	0	0	0	0
Focal segmental sclerosing glomerulopathy	0.0847	0.0469	0.1569	0.1071	0.062	0.073	0	0	0	0
Other	0.3521	0.3494	0.0308	0.381	0.3771	0.356	0.3884	0.4038	0.4031	0.3915
Mean	0.2555	0.249	0.1543	0.2533	0.2367	0.2589	0.1123	0.1452	0.1407	0.1111

Disease	**#11**	**#12**	**#13**	**#14**	**#15**	**#16**	**#17**	**#18**	**#19**	**#20**
Acute glomerulonephritis	0	0	0.0487	0.1215	0.0625	0.075	0.0635	0	0.0385	0.0215
Lupus nephritis	0.045	0.037	0.7315	0.7417	0.3806	0.6694	0.696	0.1493	0.5664	0.6259
Tubule-Interstitial nephritis	0.765	0.618	0.7206	0.6942	0.6705	0.7352	0.7088	0.6	0.7267	0.6872
IgA nephropathy	0.0407	0.1195	0.6028	0.3809	0.571	0.5667	0.36	0.3814	0.5688	0.3846
Mcd	0.0387	0.0021	0.5675	0.4237	0.067	0.5128	0.4692	0.1646	0.4754	0.4051
Fsgs	0.0913	0.1512	0.5944	0.4599	0.4498	0.547	0.4242	0.4323	0.5054	0.4388
Mpgn Type I	0	0	0	0	0	0	0	0	0	0
Mpgn Type II	0	0	0.3667	0.3505	0.1224	0.2188	0.2667	0.0379	0.2178	0.3514
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0	0	.6456	0.5856	0.0737	0.6375	0.5875	0.01617	0.2649	5243
Fibrillary glomerulonephritis	0	0	0.7475	0.6415	0.14634	0.6392	0.729	0.0625	0.5893	0.6531
Amyloidosis AA	0	0	0.2667	0.421	0	0.0889	0.4167	0.0295	0.1724	0.4074
Amyloidosis AL	0.0004	0.005	0.5882	0.497	0.19	0.4521	0.4824	0.0684	0.4309	0.5
Immunotactoid glomerulopathy	0	0	0	0.2609	0	0	0.4444	0	0.067	0.2857
Pauci-immune vasculitis	0	0	0.1351	0.1053	0.0449	0.1379	0.0457	0.0584	0.0838	0.0278
Anti-gbm nephritis	0	0	0.1714	0.0816	0.0409	0.0571	0.2381	0.0157	0.0976	0.2143
Postinfectious glomerulonephritis	0	0	0.0755	0.0784	0.0267	0	0.0822	0.0144	0.0184	0.404

Disease	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20
Lcdd	0	0.0087	0.5158	0.4511	0.3565	0.4475	0.3915	0.1058	0.3239	0.3686
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0	0.01	0.4241	0.1514	0.2701	0.4354	0.1273	0.1223	0.3927	0.1368
Hereditary nephritis	0	0	0.1935	0.25	0	0	0.3214	0.0089	0.0211	0.1754
Lhcdd	0.0021	0	0	0	0	0	0	0	0	0
Membranous nephropathy	0.1502	0.074	0.8	0.7199	0.4413	0.768	0.6849	0.3143	0.7285	0.7037
Mesangiocapillary glomerulonephritis	0	0.0004	0.5286	0.4528	0.131	0.4364	0.3896	0.035	0.474	0.4167
Anca glomerulonephritis	0.0462	0.0354	0.6205	0.4796	0.3291	0.5754	0.4565	0.2143	0.4986	0.4752
Acute tubular necrosis	0	0	0.2899	0.0677	0.0222	0.0116	0.1	0.053	0.08947	0.1027
Cast nephropathy	0	0.0012	0.439	0.2927	0.11	0.2683	0.18	0.0334	0.0818	0.2576
Tubulopathy	0.0041	0.0133	0.3338	0.2098	0.1574	0.2429	0.2073	0.22	0.2493	0.23
C3 glomerulopathy	0	0	0.4286	0.2564	0	0.0377	0.2727	0.0077	0.0376	0.1867
Amyloidosis (No type)	0	0	0.1159	0.1584	0	0.0845	0.0879	0.0345	0.0679	0.0381
Focal segmental sclerosing glomerulopathy	0	0.0004	0.0377	0.0131	0.1008	0.092	0.0179	0.0182	0.0319	0.0656
Other	0.0419	0.0016	0.3313	0.162	0.0569	0.3268	0.1723	0.2496	0.3059	0.1904
Mean	0.0409	0.036	0.3774	0.317	0.1607	0.3021	0.3141	0.1149	0.2709	0.3093

Disease	**#21**	**#22**	**#23**	**#24**	**#25**	**#26**	**#27**	**#28**	**#29**	**#30**
Acute glomerulonephritis	0.042	0.0267	0.0126	0	0.0513	0.0308	0	0.019	0.0208	0.0405
Lupus nephritis	0.3538	0.7194	0.7218	0.4424	0.6612	0.7173	0.164	0.6423	0.6963	0.4115
Tubule-Interstitial nephritis	0.6343	0.7279	0.6974	0.7019	0.7365	0.7048	0.6378	0.7285	0.6954	0.6784
IgA nephropathy	0.5556	0.6027	0.3944	0.5459	0.5559	0.3704	0.387	0.6442	0.3762	0.558
Mcd	0	0	0	0	0	0	0	0	0	0
Fsgs	0.4406	0.5993	0.4479	0.4551	0.559	0.4409	0.4294	0.5994	0.4466	0.4691
Mpgn Type I	0	0	0	0	0	0	0	0	0	0
Mpgn Type II	0.0488	0.3279	0.3478	0.1091	0.2154	0.2973	0.0211	0.1414	0.338	0.0417
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0.0952	0.5694	0.5484	0.0727	0.5678	0.5422	0.0025	0.2555	0.5672	0.0991
Fibrillary glomerulonephritis	0.1477	0.7647	0.6852	0.0625	0.6939	0.6795	0.0602	0.6	0.6667	0.1163
Amyloidosis AA	0.0321	0.3111	0.5	0	0.08	0.4615	0.3026	0.2069	0.4314	0.0395
Amyloidosis AL	0.0855	0.5695	0.5767	0.25	0.4648	0.5868	0.0726	0.5158	0.5028	0.0824
Immunotactoid glomerulopathy	0	0	0.2353	0	0	0.1667	0	0	0.0584	0
Pauci-immune vasculitis	0.1559	0.0351	0.1138	0.0182	0.0811	0.0343	0.0592	0.0632	0.4	0.145
Anti-gbm nephritis	0.0157	0.1143	0.1154	0	0.0588	0.1905	0.0081	0.2029	0.1667	0
Postinfectious glomerulonephritis	0.0485	0.08	0.0784	0.0408	0	0.0845	0.012	0.00087	0.036	0.0396

Disease	**#21**	**#22**	**#23**	**#24**	**#25**	**#26**	**#27**	**#28**	**#29**	**#30**
Lcdd	0.2103	0.5182	0.4428	0.3333	0.4519	0.3933	0.9542	0.4369	0.3504	0.2
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0.3686	0.4394	0.134	0.262	0.4644	0.1045	0.1032	0.4808	0.1513	0.3581
Hereditary nephritis	0	0.0645	0.1765	0	0	0.2353	0.009	0.0211	0.1695	0
Lhcdd	0	0	0	0	0	0	0	0	0	0
Membranous nephropathy	0.4463	0.8068	0.7435	0.4884	0.7825	0.7034	0.3256	0.7857	0.7354	0.4737
Mesangiocapillary glomerulonephritis	0.4463	0.8068	0.7435	0.4884	0.7825	0.7034	0.3256	0.7857	0.7354	0.4737
Anca glomerulonephritis	0.3333	0.6275	0.5126	0.3348	0.574	0.496	0.204	0.5703	0.5218	0.3503
Acute tubular necrosis	0.0632	0.2154	0.0772	0.0164	0.0126	0.0565	0.0487	0.078	0.0651	0.069
Cast nephropathy	0.087	0.4198	0.2933	0.1159	0.2857	0.1127	0.0274	0.1029	0.2174	0.1023
Tubulopathy	0.2507	0.332	0.2223	0.1461	0.2544	0.1988	0.2044	0.2798	0.2497	0.2379
C3 glomerulopathy	0.0189	0.2381	0.3014	0.0526	0.0408	0.2769	0	0.0292	0.25	0.0333
Amyloidosis (No type)	0.034	0.1212	0.2385	0	0.0556	0.1319	0.0354	0.0247	0.0721	0.0376
Focal segmental sclerosing glomerulopathy	0.0778	0.0208	0.0144	0.0588	0.0732	0.0177	0.0148	0.0237	0.0734	0.1217
Other	0.2567	0.3295	0.1693	0.0174	0.3279	0.1721	0.2638	0.326	0.1796	0.2859
Mean	0.1701	0.3531	0.3259	0.1559	0.3032	0.3035	0.1513	0.2944	0.3107	0.1776

Disease	#31	#32	#33	#34	#35	#36	#37	#38	#39	#40
Acute glomerulonephritis	0	0	0	0	0	0	0.0153	0	0.0259	0.1702
Lupus nephritis	0.771	0.7702	0.4093	0.6615	0.7585	0.1846	0.7635	0.738	0.4078	0.716
Tubule-Interstitial nephritis	0.7913	0.7826	0.7165	0.7959	0.788	0.6589	0.7958	0.7723	0.6857	0.7814
IgA nephropathy	0.5943	0.2258	0.6247	0.6673	0.1355	0.4493	0.6614	0.47	0.6214	0.6852
Mcd	0.6419	0.5674	0.0182	0.6031	0.5556	0.1661	0.6731	0.5512	0.1759	0.6411
Fsgs	0.6812	0.5168	0.4873	0.6411	0.4649	0.4449	0.6523	0.5877	0.5031	0.3262
Mpgn Type I	0	0	0	0	0	0	0	0	0	0
Mpgn Type II	0	0.6571	0	0	0.1224	0	0.1584	0.3385	0.04167	0.2
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0.6324	0.6571	0	0.4915	0.5397	0	0.3178	0.671	0.0971	0.5031
Fibrillary glomerulonephritis	0.602	0.701	0	0.5195	0.6889	0	0.7885	0.7429	0.15556	0.5208
Amyloidosis AA	0	0.303	0	0	0.25	0.0302	0	0.4091	0.035	0.1778
Amyloidosis AL	0.5556	0.6053	0.2075	0.4107	0.6053	0.069	0.6989	0.618	0.08	0.6923
Immunotactoid glomerulopathy	0	0	0	0	0	0	0	0	0	0.1429
Pauci-immune vasculitis	0	0	0	0	0	0.0541	0.0575	0	0.1538	0.1655
Anti-gbm nephritis	0	0	0	0	0	0.0081	0	0.0667	0	0.3429
Postinfectious glomerulonephritis	0	0	0	0	0	0.0122	0.0128	0	0.0408	0.0476

Disease	#31	#32	#33	#34	#35	#36	#37	#38	#39	#40
Lcdd	0.5	0.5306	0.3077	0.3311	0.4842	0.0852	0.4606	0.5902	0.2133	0.5678
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0.2652	0	0.2115	0.2184	0	0.0952	0.5758	0	0.405	0.494
Hereditary nephritis	0	0	0	0	0	0.0091	0	0.0645	0	0
Lhcdd	0	0	0	0	0	0	0	0	0	0
Membranous nephropathy	0.8881	0.7713	0.5107	0.8634	0.7389	0.3523	0.8818	0.7678	0.5	0.8366
Mesangiocapillary glomerulonephritis	0.4099	0.5758	0.0438	0.2993	0.4471	0.0031	0.5321	0.5701	0.1347	0.478
Anca glomerulonephritis	0.7048	0.528	0.3408	0.6426	0.4985	0.2126	0.694	0.6221	0.3682	0.6643
Acute tubular necrosis	0.0396	0	0	0.0268	0	0.0489	0.0906	0.0404	0.0628	0.225
Cast nephropathy	. 0.1429	0.1667	0.1	0	0.0056	0.0254	0.0604	0.25	0.1014	0.4301
Tubulopathy	0.1283	0.0207	0.063	0.0142	0.0513	0.1909	0.2119	0.143	0.2415	0.3948
C3 glomerulopathy	0	0.2791	0	0	0.1714	0	0	0.2553	0	0.1
Amyloidosis (No type)	0	0.0513	0	0	0.0513	0.0355	0.0287	0.087	0.038	0.1071
Focal segmental sclerosing glomerulopathy	0	0	0.0345	0	0	0.0051	0	0	0.075	0.1124
Other	0.4209	0.1326	0.0047	0.403	0.176	0.2611	0.46	0.1821	0.2813	0.4343
Mean	0.2923	0.2947	0.1360	0.253	0.2511	0.1134	0.3197	0.2956	0.1815	0.3652

Disease	#41	#42	#43	#44	#45	#46	#47	#48	#49	#50
Acute glomerulonephritis	0.0274	0.0241	0.1026	0	0	0.087	0.1154	0	0.2619	0.0408
Lupus nephritis	0.7004	0.749	0.7927	0.7857	0.3967	0.8032	0.749	0.4352	0.747	0.7386
Tubule-Interstitial nephritis	0.7747	0.7811	0.8034	0.7983	0.7296	0.8242	0.7814	0.7362	0.7408	0.7029
IgA nephropathy	0.6464	0.6754	0.7428	0.6929	0.5639	0.741	0.6704	0.5683	0.7242	0.424
Mcd	0.597	0.6088	0.677	0.613	0	0.7026	0.6418	0	0.6549	0.4516
Fsgs	0.5594	0.6152	0.7009	0.5892	0.4606	0.6921	0.6078	0.4389	0.6702	0.469
Mpgn Type I	0	0	0	0	0	0	0	0	0	0
Mpgn Type II	0.349	0.2424	0.5075	0.4719	0	0.3214	0.5063	0	0.4286	0.4557
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0.5375	0.6503	0.6711	0.6706	0.0339	0.698	0.6215	0.0222	0.667	0.6441
Fibrillary glomerulonephritis	0.6022	0.6667	0.6591	0.6038	0	0.6869	0.7048	0.0345	0.7312	0.6496
Amyloidosis AA	0.2105	0.3636	0.1951	0.3333	0.086	0.2051	0.2857	0	0.1714	0.56
Amyloidosis AL	0.5963	0.6145	0.6667	0.6667	0	0.6928	0.6784	0.1031	0.6883	0.5217
Immunotactoid glomerulopathy	0	0	0.1818	0.1429	0	0	0	0	0	0.1333
Pauci-immune vasculitis	0.0522	0.125	0.2791	0.2162	0	0.1618	0.0719	0	0.1679	0.061
Anti-gbm nephritis	0.1379	0.1026	0.2308	0.1538	0	0.2	0.2632	0	0.1379	0.186
Postinfectious glomerulonephritis	0.2	0	0.1395	0.2308	0	0	0.16	0	0.25	0.058

Disease	**#41**	**#42**	**#43**	**#44**	**#45**	**#46**	**#47**	**#48**	**#49**	**#50**
Lcdd	0.505	0.5316	0.6239	0.5492	0.2278	0.625	0.5528	0.2179	0.6636	0.4576
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0.422	0.495	0.5169	0.4881	0.228	0.6131	0.4869	0.2653	0.4706	0.1338
Hereditary nephritis	0.1333	0.1212	0.0769	0.3721	0	0.2	0.1463	0	0.3429	0.2105
Lhcdd										
Membranous nephropathy	0.609	0.8066	0.8967	0.8198	0.4571	0.8932	0.834	0.456	0.8755	0.7309
Mesangiocapillary glomerulonephritis	0.4906	0.5045	0.6047	0.5333	0.0159	0.5797	0.5	0.0469	0.5278	0.4596
Anca glomerulonephritis	0.6167	0.6511	0.7398	0.6659	0.186	0.7604	0.6705	0.1851	0.7322	0.5351
Acute tubular necrosis	0.1167	0.1176	0.2329	0.1463	0	0.2788	0.2073	0	0.2595	0.0583
Cast nephropathy	0.3256	0.3043	0.5	0.2667	0.037	0.4595	0.4211	0	0.5952	0.2418
Tubulopathy	0.3235	0.3642	0.4333	0.3663	0.0258	0.4443	0.3887	0.0259	0.3291	0.2321
C3 glomerulopathy	0.2041	0.2642	0.2273	0.386	0	0.1463	0.4138	0	0.381	0.2727
Amyloidosis (No type)	0	0.0702	0.0816	0.1	0	0.08	0.2254	0	0.1481	0.1818
Focal segmental sclerosing glomerulopathy	0.1852	0.0909	0.1299	0.2333	0	0.1249	0.0606	0	0.2381	0.0204
Other	0.4509	0.427	0.497	0.4433	0	0.4718	0.4409	0	0.3619	0.18
Mean	0.3457	0.36557	0.427	0.4064	0.1132	0.4158	0.4071	0.1161	0.4334	0.3270

Disease	**#51**	**#52**	**#53**	**#54**	**#55**	**#56**	**#57**	**#58**	**#59**	**#60**
Acute glomerulonephritis	0.021	0.0226	0.031	0.0408	0	0.022	0	0.0156	0.0271	0.0246
Lupus nephritis	0.4486	0.7004	0.6975	0.3606	0.7925	0.7371	0.4173	0.4123	0.403	0.1233
Tubule-Interstitial nephritis	0.7153	0.7449	0.7	0.6677	0.7636	0.725	0.7326	0.758	0.6998	0.2598
IgA nephropathy	0.611	0.6171	0.4269	0.5826	0.7384	0.5084	0.5822	0.6921	0.4742	0.3405
Mcd	0.0063	0.597	0.4604	0.1647	0.6541	0.5045	0	0.5376	0.3024	0.1179
Fsgs	0.4746	0.66	0.4949	0.4941	0.7032	0.5287	0.4423	0.6805	0.553	0.382
Mpgn Type I	0	0	0	0	0	0	0	0	0	0
Mpgn Type II	0,0287	0,1369	0,2381	0,0608	0.0488	0.3279	0	0.0262	0.0574	0.0162
Mpgn Type III	-	-	-	-	-	-	-	-	-	-
Hsp	0.0658	0.1377	0.4635	0.0919	0.5625	0.631	0	0.0614	0.0747	0.0392
Fibrillary glomerulonephritis	0.0889	0.6792	0.6557	0.1256	0.4737	0.7158	0	0.0613	0.1063	0.0228
Amyloidosis AA	0	0.209	0.4286	0.0271	0	0.4	0	0.0044	0.1238	0.0115
Amyloidosis AL	0.2205	0.446	0.5437	0.0898	0.6	0.6135	0.2222	0.0727	0.1533	0.0365
Immunotactoid glomerulopathy	0	0	0.0556	0	0	0.2667	0	0	0.0052	0.0037
Pauci-immune vasculitis	0.0611	0.0595	0.1095	0.1184	0.0377	0.0548	0	0.0471	0.0486	0.0378
Anti-gbm nephritis	0	0.0506	0.0777	0.0221	0	0.3556	0	0.002	0.0477	0.0095
Postinfectious glomerulonephritis	0	0.0317	0.1429	0.0302	0	0.1154	0	0.0058	0.0176	0.312

Disease	**#51**	**#52**	**#53**	**#54**	**#55**	**#56**	**#57**	**#58**	**#59**	**#60**
Lcdd	0.3316	0.3418	0.3039	0.2	0.5758	0.5617	0.2927	0.1514	0.076	0.0512
Hcdd	-	-	-	-	-	-	-	-	-	-
Diabetic nephropathy	0.3577	0.367	0.122	0.38	0.3587	0.2687	0.1919	0.1603	0.0977	0.068
Hereditary nephritis	0	0.0553	0.1714	0.0172	0	0.2439	0	0.0026	0.022	0.0095
Lhcdd	0	0	0	0	0	0	0	0	0	0
Membranous nephropathy	0.5039	0.781	0.723	0.4704	0.8943	0.7996	0.4495	0.8881	0.6866	0.296
Mesangiocapillary glomerulonephritis	0.0947	0.328	0.3036	0.1586	0.5146	0.4885	0.0157	0.0804	0.0909	0.0511
Anca glomerulonephritis	0.3497	0.6624	0.5283	0.3446	0.75	0.6165	0.290	0.7191	0.4913	0.1716
Acute tubular necrosis	0	0.0971	0.11	0.086	0.0531	0.0986	0	0.0576	0.042	0.0395
Cast nephropathy	0.1081	0.0861	0.25	0.0847	0.25	0.2609	0	0.0236	0.029	0.0217
Tubulopathy	0.1085	0.3075	0.2031	0.2742	0.3263	0.1791	0.0335	0.2746	0.3286	0.245
C3 glomerulopathy	0.0357	0.0444	0.2703	0.0377	0	0.2609	0	0.0039	0.0192	0.0124
Amyloidosis (No type)	0	0.0409	0.1584	0.278	0	0.1791	0	0.0074	0.0187	0.0156
Focal segmental sclerosing glomerulopathy	0.1099	0.0368	0.0537	0.1237	0	0.0482	0	0.0122	0.026	0.0234
Other	0	0.3629	0.1896	0.2543	0.401	0.1744	0	0.3902	0.1737	0.1564
Mean	0.1747	0.2877	0.2871	0.1755	0.3166	0.3628	0.1223	0.2049	0.1732	0.0967

Disease	#61	#62	#63
Acute glomerulonephritis	0.0194	0.0354	0.0417
Lupus nephritis	0.7189	0.6949	0.3854
Tubule-Interstitial nephritis	0.7606	0.7153	0.673
IgA nephropathy	0.6953	0.4068	0.5579
Mcd	0.6393	0.4577	0.1678
Fsgs	0.6736	0.5227	0.4858
Mpgn Type I	0	0	0
Mpgn Type II	0.0638	0.3457	0.0275
Mpgn Type III	-	-	-
Hsp	0.3186	0.5929	0.0958
Fibrillary glomerulonephritis	0.6857	0.7091	0.1364
Amyloidosis AA	0	0.4706	0.0318
Amyloidosis AL	0.625	0.5806	0.0899
Immunotactoid glomerulopathy	0	0.1429	0
Pauci-immune vasculitis	0.0347	0.0702	0.1215
Anti-gbm nephritis	0	0.1277	0.0237
Postinfectious glomerulonephritis	0.0047	0.1111	0.0248

Disease	**#61**	**#62**	**#63**
Lcdd	0.4057	0.439	0.1686
Hcdd	-	-	-
Diabetic nephropathy	0.5094	0.1705	0.3807
Hereditary nephritis	0	0.1277	0.0097
Lhcdd	0	0	0
Membranous nephropathy	0.8794	0.7534	0.4698
Mesangiocapillary glomerulonephritis	0.4641	0.4924	0.1459
Anca glomerulonephritis	0.7157	0.5473	0.3509
Acute tubular necrosis	0.0694	0.0566	0.0686
Cast nefropathy	0.0556	0.2885	0.0923
Tubulopathy	0.2764	0.2189	0.2642
C3 glomerulopathy	0	0.2712	0.0333
Amyloidosis (No type)	0	0.2316	0.0375
Focal segmental sclerosing glomerulopathy	0.0095	0.0755	0.0966
Other	0.3981	0.1852	0.2622
Mean	0.3008	0.328	0.1749

Table 16: k-fold model evaluation; F1 score of the 63 models,