

Labelling Clinical Reports by Exploiting the Data Structure Through Topic Modelling and Active Learning

Uppmärkning av kliniska rapporter genom att utnyttja strukturen hos datan med topic modeller och active learning

Simon Lindblad

Supervisor : Marco Kuhlmann
Examiner : Arne Jönsson

External supervisor : Mikael Nilsson

Upphovsrätt

Detta dokument hålls tillgängligt på Internet – eller dess framtida ersättare – under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår. Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns lösningar av teknisk och administrativ art. Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart. För ytterligare information om Linköping University Electronic Press se förlagets hemsida <http://www.ep.liu.se/>.

Copyright

The publishers will keep this document online on the Internet – or its possible replacement – for a period of 25 years starting from the date of publication barring exceptional circumstances. The online availability of the document implies permanent permission for anyone to read, to download, or to print out single copies for his/hers own use and to use it unchanged for non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional upon the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility. According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement. For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: <http://www.ep.liu.se/>.

Abstract

Abstract.tex

Acknowledgments

Acknowledgments.tex

Contents

Abstract	iii
Acknowledgments	iv
Contents	v
List of Figures	vi
1 Introduction	1
1.1 Motivation	2
1.2 Aim	2
1.3 Research questions	3
1.4 Delimitations	3
1.5 Structure of the Report	3
2 Theory	4
2.1 Text Processing using Unsupervised Techniques	4
2.2 Text Classification	8
2.3 Active Learning	9
2.4 Related Work	11
3 Method	12
4 Results	13
5 Discussion	14
5.1 Results	14
5.2 Method	14
5.3 The work in a wider context	15
6 Conclusion	16
Bibliography	17

List of Figures

2.1	Diagram of the LDA model.	5
2.2	(a) to (e) shows iterations of K-means until convergence. In (e) it can be seen that the new centroids capture the same documents as the previous iteration, and we have converged.	8
2.3	An overview of an active learning system	10



1 Introduction

The world's population is growing each year. Making healthcare more efficient and robust is of great importance in order to handle the challenges that arise with a growing population. One way of increasing the efficiency as well as the quality of healthcare is to create automated systems that can aid doctors in their process. As the population is growing it's of utmost importance to ensure that the quality of diagnosis remains high, and to minimize the risk of missing some critical piece of information. Taking advantage of the available medical information is key to creating aforementioned systems.

Information pertaining to a patient's diagnosis is often in the form of written clinical reports. One example where this information could be utilized is when a doctor is writing such reports. If a system could show cases with similar features as the current one, the doctor could compare the findings and check if they have obtained an abnormal result. Being able to perform such a comparison will result in extra quality assurance in the diagnostic flow. It could also provide doctors with extra confidence in that their diagnosis is correct.

The problem systems like this would face is to identify the type of a medical report in order to make further suggestions. One approach that is commonly used for such problems is machine learning. In machine learning, you use a set of inputs and map it to some output values. This is done by using data to build a, usually statistical, model.

The task of predicting a type, or class, for a given text document is called text classification. Text classification is usually solved using supervised learning. In supervised text classification, you have a set of inputs, in this case text data, that already has a category assigned to it. This data is then used to fit the model so that it later make predictions for inputs that it has not yet been exposed to. A model that have been shown to be successful in text classification is Support Vector Machines (SVM).

In order to assign fit a machine learning model to predict categories for clinical reports, we need a set of already labeled data. That is, we need to assign categories to the existing set of clinical reports. It is often the case that text data is widely available, but it is harder to come by data that is already labeled. Obtaining high quality data is important to use in machine learning systems, both in healthcare and other areas. Since the models require a sufficient amount of reports to be labeled, the task of labeling them can be cumbersome. Especially in the case of clinical data, since doctors and other clinicians time is valuable and expensive. By improving the process and the quality of data to be labeled, they can spend more time doing their job.

The field within machine learning that is focused on the task of labeling data is called active learning. It is a form of semi-supervised learning. The algorithm queries an oracle (in this case a doctor) for labels for the data points that it think will help the model improve the best. This is used when there is plenty of readily available data, but assigning labels is expensive. Since the data points to be labeled are actively selected, the models can require fewer examples than if they were selected at random. The points can be selected by considering the certainty of the models, and request to label the documents that the model is less certain about. Another approach, which has not been given as much attention, is using the underlying structure of the data to select points. The goal with this approach is that you can capture the distribution of the categories.

If you assign one of two classes to each document, you have a binary classification problem. Problems where you assign one of several classes is called a multi-class classification problem. Multi-labeled classification is when you assign one or more label to each document. It type of classification that will be treated in this report is multi-labeled. Assigning several classes to a document is more time consuming than in the cases where you only need to find one option. In those cases you can stop when you have found the appropriate label. However, when a document can be assigned several classes you need to consider the entire report. This makes the use of active learning methods to enhance the labeling of documents even more useful in the multi-labeled case.

1.1 Motivation

This thesis is carried out at Sectra Medical Imaging IT Solutions AB, as a part of their research group. They are currently pursuing a research project with Region Skåne in southern Sweden. The intention behind the project is to use machine learning techniques to, among other things, be able to suggest categories to doctors while they are writing medical reports. Another case is to use the categories of documents to present doctors with medical reports that are handling similar cases from the past. With this information the doctors could get an extra quality assurance check in their diagnostic flow.

In order to build these system, you need a substantial amount of labeled clinical reports. Therefore, the purpose of this thesis is to increase the quality and efficiency of labeling these reports. This will be done by using unsupervised learning techniques such as topic models and clustering to first remove documents that aren't supposed to be labeled. That includes documents that describe patients never showing up for a scan, deceased patients and patients being moved to a different hospital, among others. For the labeling, a system is be built to use active learning in conjunction with the aforementioned unsupervised techniques to increase the quality of the labeled documents. In the work that they have done so far, the doctor that primarily worked with the labeling of reports stated that the distribution over the labeled categories are very skewed. The vast majority of labeled documents were assigned the same few categories. This in turn leads the models to require a lot of labeled samples to work well. By using active learning techniques we can reduce the number of labeled samples needed to obtain an accurate model.

1.2 Aim

The purpose with this thesis project is to evaluate different solutions to increate the quality of labeled reports, and thereby reducing the amount of them needed for a system. Resulting from this will be a complete, standalone system, for labeling reports. The reports are interactively queried so a user can label the reports that are deemed most useful by the system.

1.3 Research questions

The specific research questions that this thesis will treat is presented here. They will be the main focus of study.

1. *Is it possible filter out invalid clinical reports by using unsupervised techniques such as topic models and clustering?*

In the dataset from Sectra, there are reports describing patients not showing up for or changing the time of their appointments, deceased patients or patients that have been ordered to another hospital. These reports does not contain any information of value from a medical point of view and should not be considered in the report labeling process.

Unsupervised machine learning models such as topic modeling and clustering does by definition not require any labeled documents to train on. If it is possible to, without any such data, group these invalid reports together and remove them from the process before a doctor is presented with them that would be an additional hurdle removed from the process.

2. *What are the alternatives to sample documents at random in a document labeling system?*

How we are choosing the documents to be sampled is important. If the dataset that is being sampled is skewed, i.e. some categories are a more frequent than others, our labeled set will likely follow that distribution. This will result in the system requiring a lot of labeled documents to gain a high accuracy with reports of less common categories.

If the decision boundaries of our models can be used to pick documents that would be more informative, the number of labeled documents could be reduced and still gain the same accuracy. Another approach to selecting the documents to sample is to take advantage of the underlying structure of the data through clustering.

3. *Which of the alternatives in question 2 gives us the highest quality set of labeled reports?*

The algorithms to be evaluated will be from the two approaches: based on the model certainty or the underlying structure of the data (or both). When choosing the algorithm to use, there are several different factors that will affect the final results and therefore needs to be taken into consideration. How well the models perform on the data is a rather obvious one – evaluating the models based on accuracy, precision, recall and f1-score. But they also need to be able to query documents in a reasonable time, if it is expensive to label reports it is likely to be expensive to wait for the reports to be queried. Choosing reports in batches and if the algorithm needs a big initial set of labeled reports are other factors that will be evaluated.

1.4 Delimitations

To be written - data and evaluation by only one doctor

1.5 Structure of the Report

To be written



2 Theory

In this section the theory behind the techniques used during the thesis work will be presented. The first part will go through techniques used to process the data and perform an exploratory analysis. After that text classifications, primarily with SVM, will be covered. The last section contains an overview of the field of active learning, as well as a comparison of some different active learning techniques for multi-labeled data.

2.1 Text Processing using Unsupervised Techniques

Techniques in machine learning that does not require you have a categorized or labeled set of data is called unsupervised. They use the structure of the data to obtain the information to use when processing it. When it comes to text data, there are a few common methods and techniques that are unsupervised, and can be used for different purposes. Examples of such techniques are *topic modeling* and *clustering*. Another interesting technique is *word2vec*, that is used to produce word embeddings.

When working with text, it needs to be represented in a way that allows the models to work with it effectively. *Bag-of-words* (BoW) is one of the more common representations when performing text analysis. Using BoW the text is represented as a multi-set. That is, a document is represented by the number of occurrences of the different words. The representation of a document therefore becomes very high-dimensional, there is one dimension for each word in the vocabulary. Like the name implies, the positions of the words are not taken into account, they are viewed as if they were taken from a bag. Another drawback from this approach is that a word in a written language can be used to express several different thoughts, and one thought can be expressed using several different words. However, it is easy to work with, and is used when performing topic modeling among other things.

One way to incorporate positional information into the representation is the use of *n*-grams. Instead of storing information pertaining to one term, information is stored with regards to *n* consecutive terms. Considering the text "Pattern Recognition and Machine Learning" using a bigram (*n*-gram with $n=2$), would result in the tokens: "Pattern Recognition", "Recognition and", "and Machine", and "Machine Learning".

Topic Modeling

A topic model is a statistical model for finding topics within text [7]. The topics build upon the probability that a certain word would occur in a text about a given topic, on the basis of terms occurring together. For example, if the topic represents United States politics, words such as “government”, “Trump”, “Reagan”, “Senate”, or “Medicaid” are more likely to appear than “sailboat” or “sweater”. Any given document can then contain a topic with some probability. This can be viewed as fuzzy clustering, and that the document has a degree of membership in a topic or cluster [7]. The most common topic model in use is Latent Dirichlet Allocation (LDA) [7]. Another topic model that preceded LDA is Probabilistic latent semantic analysis (PLSA) [10]. However, PLSA has been shown to be more prone to overfitting than LDA [7].

In the rest of the report, the following notation will be used:

- D denotes a corpus of M documents: $D = \{w_1, w_2, \dots, w_M\}$.
- The number of topics is K . Each topic is indexed by i .
- N_d is the number of terms in document d .
- N_i is the number of terms in topic i .
- V denotes the number of words in the vocabulary.

Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) is a statistical model, where abstract topics in the model are defined as distributions over words [5]. LDA is based on a generative process, a model of which can be seen in Figure 2.1. The circles in this figure represent random variables. Dependencies between these random variables are shown with arrows, and if a variable is observed it is shaded in the figure. In this model, the only observed variable is the words in the document. Parts of the model are surrounded by a rectangle to show that the part is repeated several times.

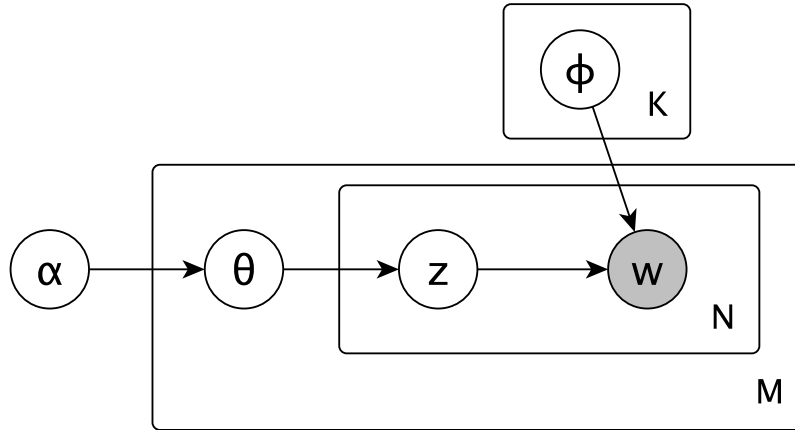


Figure 2.1: Diagram of the LDA model.

The generation of a corpus is done with the following steps [7, 5]:

- **Draw a distribution over the words for each topic.** A sample ϕ_i is drawn from a symmetric Dirichlet distribution with parameter β . This sample represents the distribution of terms for the topic i .

$$\Phi_i \sim \text{Dir}(\beta) \quad (2.1)$$

$$p(\Phi_i|\beta) = \frac{\Gamma(V\beta)}{\Gamma(\beta)^V} \prod_{v=1}^V \phi_{iv}^{\beta-1} \quad (2.2)$$

Here, Γ is the gamma function.

- **Draw a distribution over the topics for each document.** A sample θ_d is drawn from a Dirichlet distribution with parameters α . This sample represents the distribution of topics for document d .

$$\Theta_d \sim \text{Dir}(\alpha) \quad (2.3)$$

$$p(\Theta_d|\alpha) = \frac{\Gamma(\sum_{i=1}^K \alpha_i)}{\prod_{i=1}^K \Gamma(\alpha_i)} \prod_{i=1}^K \theta_{di}^{\alpha_i-1} \quad (2.4)$$

- For each token with index n :
 - **Draw a topic assignment z_{dn} for the token index n .** z_{dn} is drawn from the distribution over topics for each document. That is, z_{dn} is drawn from a multinomial distribution using θ_d as a parameter.

$$z_{dn} \sim \text{Multinomial}(\Theta_d) \quad (2.5)$$

$$p(z_{dn} = i|\Theta_d) = \theta_{di} \quad (2.6)$$

- **Draw a token w_{dn} .** The token w_{dn} is drawn from the topic distribution assigned to the index n . That is, w_{dn} is drawn from a multinomial with parameter $\phi_{z_{dn}}$.

$$w_{dn} \sim \text{Multinomial}(\Phi_{z_{dn}}) \quad (2.7)$$

$$p(w_{dn} = v|z_{dn} = i, \Phi_i) = \phi_{iv} \quad (2.8)$$

The LDA model identifies topics from different terms that occur together. Consider the case where an LDA model has been used to learn a number of topics. Two terms that frequently occur together are then likely to be in the same topic. So, if the same word has been used to express different thoughts, and the word has the same probability in two topics, the words that it co-occurs with can be used to differentiate between the different thoughts.

The task of learning the LDA model is a Bayesian Inference problem. We have several variables that we cannot observe: the word distribution for the topics (ϕ_i), the topic assignments for the tokens (z), and the topic distribution for the documents (θ_d). The only observed variables are the words in the document. We have to approximate the posterior distribution using some sampling method, since it cannot be inferred automatically [5].

There exist a few algorithms that can be used to learn topics for the LDA model. Two of these that has shown to be able to extract useful topics from text are *collapsed Gibbs sampling* [9] and *variational Bayes* [5]. In collapsed Gibbs sampling, θ and ϕ are marginalized out.

It works by repeatedly sampling the topic assignment z_{dn} for each token, conditioned on the assignments for the other tokens. Variational Bayes works by using simpler single-variable models to approximate the LDA. As a consequence, it disregards any dependencies between the variables. This is the approach used in the original LDA paper [5].

Collapsed Gibbs

To be written

Text Clustering

Cluster analysis is commonly defined as finding groups in a given dataset. The members of these groups are determined to be similar by a similarity measure [11, 2]. Since text data is sparse, but yet have a very high dimensionality. With one dimension per term in the dictionary, it is not uncommon with dimensions in the order of 10^5 . For this reason, some of the more naive clustering algorithms does not work as well for text data [2].

In distance-based clustering, a similarity function is used to measure the closeness between two text documents. For the purpose of measuring the similarity between text objects, the cosine similarity function is most commonly used [2]. Two different approaches to distance-based clustering are distance-based partitioning, and agglomerative hierarchical clustering.

K-means Clustering

When using the k-means clustering algorithm, the clusters are based upon an initial set of k representatives. A simple approach to k-means clustering can be seen as:

1. Select K seeds from the original dataset
2. Assign the rest of the documents to one of these seeds, based how how similar they are by the similarity function
3. Before each new iteration, select a new centroid for each cluster. This should be the point that is the best central point for the cluster.
4. Repeat step 2 and 3 until convergence.

A visualization of this can be seen in Figure 2.2. One advantage that K-means has over K-medoid is that it requires a small number of iterations, especially compared to K-medoid [2, 15]. However, K-means is rather sensitive to the selection of initial seeds. One approach is to just select them randomly, or selecting them based on the result of another lightweight clustering method. A frequently used method is k-means++, that has been shown to improve both the speed and accuracy of k-means clustering [3].

Hierarchical Clustering

To be written if used in the end

Word Embeddings with Word2Vec

To be written

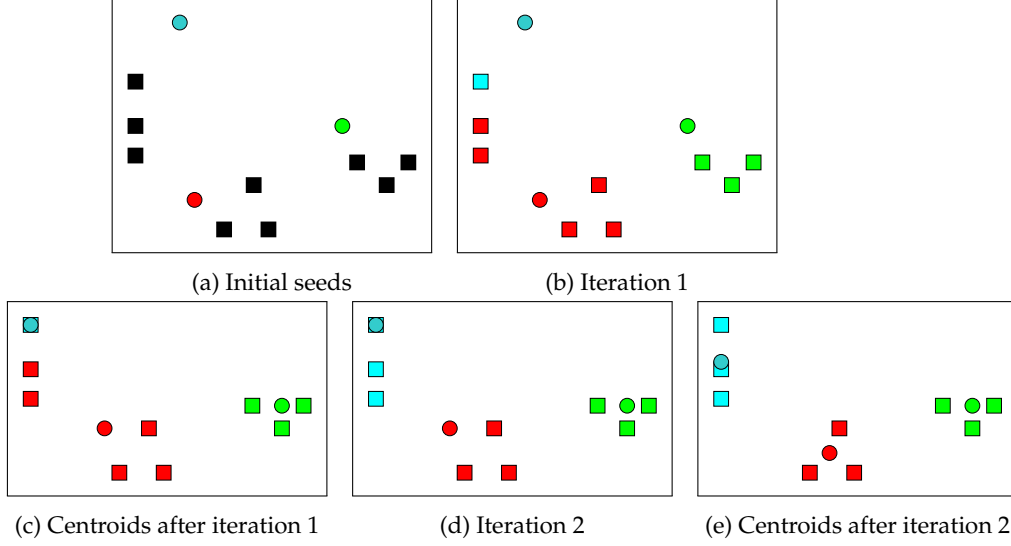


Figure 2.2: (a) to (e) shows iterations of K-means until convergence. In (e) it can be seen that the new centroids capture the same documents as the previous iteration, and we have converged.

2.2 Text Classification

Text classification is a widely studied field within Computer Science. It is an important problem in supervised machine learning, and it is the task of assigning one or more classes to a given text document [1]. The problem is mainly approached with supervised machine learning. That is, with a dataset that consists of a collection of text documents, where each document has one or more classes assigned to it. With the help of these labels, a classification model is fitted to the data. The goal of this is for the model to be able to correctly assign a class to a previously unseen text document. Some of these classification models can also produce a probability of a document being of a certain class. Other models are based on the concept of a margin that separates the classes, and the distance between a data point and a margin can be used to indicate how certain the model is of the assigned label [17]. Example of use cases for text classification is categorization of news articles, document retrieval and email filtering. There exists several different models for classifying text. Decision trees, neural networks and Support Vector Machines (SVM) are some have been previously applied to the text domain with successful results [2]. In this thesis, SVM are the main focus, since they have been studied extensively in the context of active learning.

Support Vector Machines

SVMs work by implicitly map the training data to a feature space [4]. The goal is that the data should be linearly separable in the feature space, even if it is not in the input space. In the case of binary classification, a point is classified by the linear model:

$$y(x) = w^T \phi(x) + b \quad (2.9)$$

The sign of $y(x)$ determines the label assigned to x .

SVMs work by trying to find the hyperplane that maximizes the margin. That is, the distance between any point and the decision boundary should be as large as possible. The hyperplane that gives us the maximum margin can be found by:

$$\arg \min_{w,b} \frac{1}{2} \|w\|^2 \quad (2.10)$$

In order to allow for better generalization, and for data that isn't completely linearly separable, SVMs make use of slack variables ξ_n to penalize points that are close to the decision boundary [4]. A parameter $C > 0$ controls how much effect the slack variables will have. The equation with the slack variables becomes:

$$\arg \min_{w,b} \frac{1}{2} \|w\|^2 + C \sum_n \xi_n. \quad (2.11)$$

A smaller C -value allows more points to be misclassified, in order to achieve better generalization.

Multi-Label Classification

Multi-label classification is the type of text classification where one instance can be associated with multiple labels. It is a generalized version of the multi-class classification problem, where you have more than 2 labels, but each document is only assigned one [18].

A common way of solving multi-label classification problems is the Binary Relevance method [14, 6]. It is a way of transforming the multi-label classification problem into several different binary ones. With Binary Relevance you fit one classification model per label in your data. Each of these classifiers are then predicting whether or not the document is associated with the corresponding label or not.

2.3 Active Learning

Conventional machine learning systems use a set of available data to find a hypothesis that can explain the patterns. The purpose of active learning is to allow a system to *select* the data that it wants labeled, and therefore the data it wants to be trained on [16]. An active learning system samples a document to be labeled, and then queries an oracle (often a human annotator) to get the label for that document. By being able to decide what data to label and use, the goal is that the system can achieve better results, and that the data will be of higher quality. A model of the active learning system can be seen in Figure 2.3. The main

In several different domains, data is readily available and easy to come by. But even if the data is abundant, labels for the data is often harder and more expensive to come by [16], especially when it comes to multi-label problems.

The next section will describe different ways to access the documents in active learning systems, followed by some theory of how the samples relate to the hypothesis space. After that some concrete methods for selecting the samples to queried are described and compared.

Pool-Based Sampling

The main focus is how to select the samples to be labeled. There are different sampling methods in use, and which one is more appropriate depends on how the data can be accessed. Pool-based sampling is motivated by the assumption that there exists a large ready pool of data, where only a small portion is labeled [12, 16]. The samples to be labeled is then selecting by evaluating the entire pool of unlabeled data, and select the most appropriate one based on a defined utility measure. If the entire pool is large, a subset could be used instead. For applied active learning, pool-based sampling seems to be the most popular choice, but there are some alternatives that have been used in theoretical settings such as stream-based selective sampling. The difference between stream-based selective sampling and pool-based sampling is the individuality of the decisions in stream-based selective sampling, where you draw one sample at a time from an input source and make the decision whether or not to query a label for it. For text applications, where a set of data is often readily available, pool-based sampling is often the more appropriate option since you can consider the entire dataset. Pool-based is therefor the sampling technique that will be considered in this paper.

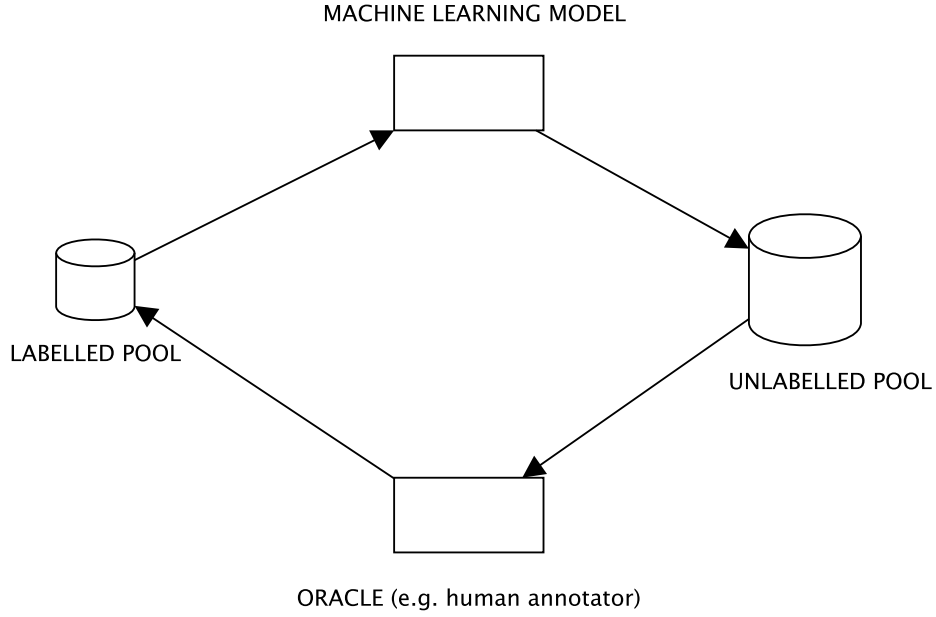


Figure 2.3: An overview of an active learning system

Initial Sampling Selection

To be written

Searching the Hypothesis Space

In machine learning hypothesis is a specific configuration of a model, the purpose of which is to predict outputs on new instances of data by generalizing the training data. One hypothesis can, for example, be an SVM model with specific values for the parameters. The set of all possible hypothesis that we are working with the *hypothesis space*. Following the SVM example, the hypothesis space would be the set of SVM's with the different values that are possible for the parameters. The hypothesis space is defined as:

$$\mathcal{H} = \left\{ f \mid f(x) = \frac{w * \phi(x)}{|w|} \right\} \quad (2.12)$$

where $w \in \mathcal{W}$ and \mathcal{W} is our parameter space

The version space the subset of the hypothesis space The subset of the hypothesis space that in the feature space separates the data is called the version space, which is defined as:

$$\mathcal{V} = \left\{ f \in \mathcal{H} \mid y_i f(x_i) > 0 \forall i \in \{1 \dots n\} \right\} \quad (2.13)$$

So the version space therefore represents the different hypothesis that make correct predictions on the training data. Under the assumption that one of the hypothesis can fully separate the data, the version space shrinks when more labeled data is acquired. So for new labeled instances the hypotheses in the version space will give better predictions for the training data. Based on this, an active learning algorithm should aim to reduce the size of the version space with each new sample, optimally make it half the size in each iteration.

There exists a useful relationship between the feature space \mathcal{F} and the hypothesis space \mathcal{H} called the *version space duality* [17, 19]. It states that hyperplanes in the hypothesis corresponds to points in the feature space, and the other way around. So by selecting points to be labeled, constraints can be enforced on the hypothesis that form the version space.

One approach to this is called that has shown to be successful is *Uncertainty Sampling* [16]. The idea behind SVM is to find a hyperplane that separates two classes in a binary classification with the maximum margin. Out of the different hyperplanes in the hypothesis space, the version space contains those that can successfully separate the data. We want to select the points in the feature space that will reduce the amount of the valid hypotheses the most. Since SVMs tries to find the support vectors that maximizes the decision boundary in the feature space, separating the two classes. Considering this in \mathcal{H} , it will be analogous to the hypothesis in the center of the hypothesis space encompassed constraints set by the labeled points. What Uncertainty Sampling is predicting the values for the unlabeled points, and then choose the one that it is most uncertain about, the one closest to the decision boundary, to be labeled. Based on the version space duality, it is a good approximation for dividing the version space in two.

Binary Minimization

Binary Minimization is a generalization of uncertainty sampling, to make it work with multi-label data. The approach taken is to decompose the multi-label problem to several binary one-vs-rest tasks, like discussed in 2.2. The unlabeled point that is chosen for labeling is then the one with the smallest SVM margin across all the binary classification tasks. By doing this, it does not incorporate the multiple labels into the decision process, but treats all classes individually and equally.

Maximum Loss Reduction with Maximum Confidence

Adaptive Active Learning

2.4 Related Work

Active learning has been researched in text classification with different approaches. They can be seen as two categories: searching through the hypothesis space by using the uncertainty of a model, or by exploiting the structure of the data through clustering [8].

One of the common baselines for active learning is uncertainty sampling [12]. That simply queries the label for the data point the model is most uncertain about. In [8] hierarchical clustering is used in an active learning system. The labels are queried from clusters where there is a lot of uncertainty when it comes to the majority label. By pruning the tree of clusters while querying for labels the goal is to obtain a pruning where each node mostly contains one label.

In [13] they also take advantage of a clustering to select the samples to be labeled in a two-class environment. They take advantage of that the data points closest to the centroids are the most important ones, and that most data points in one cluster have the same label. What this approach has in common with a lot of the current research is that it is treating single-label or binary classification problems.



3 Method

In this chapter, the method is described in a way which shows how the work was actually carried out. The description must be precise and well thought through. Consider the scientific term replicability. Replicability means that someone reading a scientific report should be able to follow the method description and then carry out the same study and check whether the results obtained are similar. Achieving replicability is not always relevant, but precision and clarity is.

Sometimes the work is separated into different parts, e.g. pre-study, implementation and evaluation. In such cases it is recommended that the method chapter is structured accordingly with suitable named sub-headings.



4 Results

This chapter presents the results. Note that the results are presented factually, striving for objectivity as far as possible. The results shall not be analyzed, discussed or evaluated. This is left for the discussion chapter.

In case the method chapter has been divided into subheadings such as pre-study, implementation and evaluation, the result chapter should have the same sub-headings. This gives a clear structure and makes the chapter easier to write.

In case results are presented from a process (e.g. an implementation process), the main decisions made during the process must be clearly presented and justified. Normally, alternative attempts, etc, have already been described in the theory chapter, making it possible to refer to it as part of the justification.



5 Discussion

This chapter contains the following sub-headings.

5.1 Results

Are there anything in the results that stand out and need be analyzed and commented on? How do the results relate to the material covered in the theory chapter? What does the theory imply about the meaning of the results? For example, what does it mean that a certain system got a certain numeric value in a usability evaluation; how good or bad is it? Is there something in the results that is unexpected based on the literature review, or is everything as one would theoretically expect?

5.2 Method

This is where the applied method is discussed and criticized. Taking a self-critical stance to the method used is an important part of the scientific approach.

A study is rarely perfect. There are almost always things one could have done differently if the study could be repeated or with extra resources. Go through the most important limitations with your method and discuss potential consequences for the results. Connect back to the method theory presented in the theory chapter. Refer explicitly to relevant sources.

The discussion shall also demonstrate an awareness of methodological concepts such as replicability, reliability, and validity. The concept of replicability has already been discussed in the Method chapter (3). Reliability is a term for whether one can expect to get the same results if a study is repeated with the same method. A study with a high degree of reliability has a large probability of leading to similar results if repeated. The concept of validity is, somewhat simplified, concerned with whether a performed measurement actually measures what one thinks is being measured. A study with a high degree of validity thus has a high level of credibility. A discussion of these concepts must be transferred to the actual context of the study.

The method discussion shall also contain a paragraph of source criticism. This is where the authors' point of view on the use and selection of sources is described.

In certain contexts it may be the case that the most relevant information for the study is not to be found in scientific literature but rather with individual software developers and open

source projects. It must then be clearly stated that efforts have been made to gain access to this information, e.g. by direct communication with developers and/or through discussion forums, etc. Efforts must also be made to indicate the lack of relevant research literature. The precise manner of such investigations must be clearly specified in a method section. The paragraph on source criticism must critically discuss these approaches.

Usually however, there are always relevant related research. If not about the actual research questions, there is certainly important information about the domain under study.

5.3 The work in a wider context

There must be a section discussing ethical and societal aspects related to the work. This is important for the authors to demonstrate a professional maturity and also for achieving the education goals. If the work, for some reason, completely lacks a connection to ethical or societal aspects this must be explicitly stated and justified in the section Delimitations in the introduction chapter.

In the discussion chapter, one must explicitly refer to sources relevant to the discussion.



6

Conclusion

This chapter contains a summarization of the purpose and the research questions. To what extent has the aim been achieved, and what are the answers to the research questions?

The consequences for the target audience (and possibly for researchers and practitioners) must also be described. There should be a section on future work where ideas for continued work are described. If the conclusion chapter contains such a section, the ideas described therein must be concrete and well thought through.



Bibliography

- [1] Charu C Aggarwal and ChengXiang Zhai. "A survey of text classification algorithms". In: *Mining text data*. Springer, 2012, pp. 163–222.
- [2] Charu C Aggarwal and ChengXiang Zhai. "A survey of text clustering algorithms". In: *Mining text data*. Springer, 2012, pp. 77–128.
- [3] David Arthur and Sergei Vassilvitskii. "k-means++: The advantages of careful seeding". In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics. 2007, pp. 1027–1035.
- [4] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [5] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3. Jan (2003), pp. 993–1022.
- [6] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. "Learning multi-label scene classification". In: *Pattern recognition* 37.9 (2004), pp. 1757–1771.
- [7] Steven P Crain, Ke Zhou, Shuang-Hong Yang, and Hongyuan Zha. "Dimensionality reduction and topic modeling: From latent semantic indexing to latent dirichlet allocation and beyond". In: *Mining text data*. Springer, 2012, pp. 129–161.
- [8] Sanjoy Dasgupta and Daniel Hsu. "Hierarchical sampling for active learning". In: *Proceedings of the 25th international conference on Machine learning*. ACM. 2008, pp. 208–215.
- [9] Thomas L Griffiths and Mark Steyvers. "Finding scientific topics". In: *Proceedings of the National academy of Sciences* 101.suppl 1 (2004), pp. 5228–5235.
- [10] Thomas Hofmann. "Probabilistic latent semantic analysis". In: *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc. 1999, pp. 289–296.
- [11] Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Vol. 344. John Wiley & Sons, 2009.
- [12] David D Lewis and William A Gale. "A sequential algorithm for training text classifiers". In: *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. Springer-Verlag New York, Inc. 1994, pp. 3–12.
- [13] Hieu T Nguyen and Arnold Smeulders. "Active learning using pre-clustering". In: *Proceedings of the twenty-first international conference on Machine learning*. ACM. 2004, p. 79.

-
- [14] Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. "Classifier chains for multi-label classification". In: *Machine learning* 85.3 (2011), p. 333.
 - [15] Hinrich Schütze and Craig Silverstein. "Projections for efficient document clustering". In: *ACM SIGIR Forum*. Vol. 31. SI. ACM. 1997, pp. 74–81.
 - [16] Burr Settles. "Active learning". In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6.1 (2012), pp. 1–114.
 - [17] Simon Tong and Daphne Koller. "Support vector machine active learning with applications to text classification". In: *Journal of machine learning research* 2.Nov (2001), pp. 45–66.
 - [18] Grigorios Tsoumakas and Ioannis Katakis. "Multi-label classification: An overview". In: *International Journal of Data Warehousing and Mining* 3.3 (2006).
 - [19] Vladimir Vapnik. *Statistical learning theory*. 1998. Wiley, New York, 1998.