Expert Systems With Applications 88 (2017) 402–418

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Identifying child abuse through text mining and machine learning

Chintan Amrit[a,*], Tim Paauw[b], Robin Aly[c], Miha Lavric[a]

[a] IEBIS Department, University of Twente, Enschede, The Netherlands
[b] Ynformed, Utrecht, The Netherlands
[c] DB Schenker, Essen, Germany

ARTICLE INFO

Article history:
Received 15 November 2016
Revised 19 May 2017
Accepted 22 June 2017
Available online 5 July 2017

Keywords:
Child abuse
Decision support
Text mining
Machine learning

ABSTRACT

In this paper, we describe how we used text mining and analysis to identify and predict cases of child abuse in a public health institution. Such institutions in the Netherlands try to identify and prevent different kinds of abuse. A significant part of the medical data that the institutions have on children is unstructured, found in the form of free text notes. We explore whether these consultation data contain meaningful patterns to determine abuse. Then we train machine learning models on cases of abuse as determined by over 500 child specialists from a municipality in The Netherlands. The resulting model achieves a high score in classifying cases of possible abuse. We methodologically evaluate and compare the performance of the classifiers. We then describe our implementation of the decision support API at a municipality in the Netherlands.

1. Introduction

Child abuse is a serious problem, with an estimated 40 million children being subject to abuse or neglect each year (World Health Organization, 2001). For 2014 alone, there have been 3.62 million referrals of alleged child maltreatment in the USA, resulting in the identification of 702,000 victims (including 1580 fatalities) of child abuse and neglect. Despite these high numbers of registrations and identified victims, cases of child abuse still remain unregistered and/or unidentified, due to missing and incomplete information, preventing adequate procedures (U.S. Department of Health & Human Services, Administration for Children & Families, Administration on Children, Youth & Families, Children's Bureau, 2016). In the same year, an UK inquiry into child sexual abuse in the family environment by the Children's Commissioner showed that as little as only 1 in 8 victims of sexual abuse may come to the attention of the police and children's services, with up to two-thirds of all sexual abuse happening in and around the family (Longfield, 2015). One way to improve the registration of child abuse is by providing training to stakeholders. Indeed, a German study concluded that everyone working in the area of child protection should receive additional interdisciplinary training (Bressem et al., 2016). However, such training might prove to be costly and time-consuming.

An alternative approach is to provide child health care professionals with a decision support system, assisting them to identify cases of possible child abuse with a higher precision and accuracy. Recent research has tried to predict cases of child abuse using structured data (Gillingham, 2015; Horikawa et al., 2016). While these methods achieve a reasonable performance, they do not take the knowledge of the pediatrician into account (Goldman, 1990). One of the sources of evidence that health care professionals often create as part of their daily procedures is free-text. As such texts are less constraining than structured data, they possibly incorporate elements of doctors' tacit knowledge of the phenomena that are not included in structured data (Henry, 2006; Malterud, 2001).

In this article, we propose a decision support based approach to increase the number of correctly identified child abuse cases and improve their registration. We aim at providing health care professionals timely and appropriate decision support about possible child abuse based on patterns in health data that the health care professionals create as part of their daily procedures.

The findings of this article are validated based on data from the Netherlands, where each child visits the public health organization (GGD[1]) roughly 15 times between the ages zero and four. The pediatricians and nurses performing these consultations use information systems to keep track of each child's development. Depending on the type of consult, both structured and unstructured data are recorded, where structured data can be the child's height or weight and unstructured data consists of free-text containing

* Corresponding author.
E-mail addresses: c.amrit@utwente.nl, camrit@gmail.com (C. Amrit), tim@ynformed.nl (T. Paauw), r.aly@utwente.nl (R. Aly), m.lavric@utwente.nl (M. Lavric).

[1] GGD from its Dutch spelling: Geneeskundige en Gezondheidsdienst.

http://dx.doi.org/10.1016/j.eswa.2017.06.035
0957-4174/© 2017 Elsevier Ltd. All rights reserved.

the pediatrician's remarks during the consultation. We first explore whether these consultation data contain meaningful patterns concerning child abuse. We then investigate whether machine learning from this data can help in identifying cases of child abuse. We, therefore, train our machine learning classifiers on cases of abuse as determined by over 500 child specialists from GGD Amsterdam, the largest public health organization in The Netherlands. We perform a methodological evaluation of a wide range of methods to identify their strengths and weaknesses. We then evaluate the automatic classifications with judgments of pediatricians, and thereby address our intention to provide decision support (in identifying child abuse) for pediatricians. We finally describe our implementation of the decision support API at GGD Amsterdam.

In summary, our contributions to research and practice are: (i) Unlike previous studies, our study is based on a large dataset that is complete both in terms of quantity (all the children of the Amsterdam region over a 4 year time period) and quality (detailed information about every child included), (ii) Uniquely, this study takes both structured and unstructured data into account in building a prediction model, (iii) Moreover, in general, our model performs better than previous models (on our particular data) and we provide insight into the inner workings of the model (please see Section 5.13); and, (iv) Owing to the model's good performance it has been deemed useful for day to day operations, and we describe its implementation in a decision support system API for identifying child abuse in this study (see Section 5.15). Such an implementation, in our opinion, is a contribution to theory and practice, as it describes both the method of construction (of the underlying algorithms) and the deployment of the API.

The remainder of this paper is structured as follows: Section 2 describes related work, Section 3 presents the models and methods we utilized in our research. Section 4 describes how we pre-processed the data, Section 5 describes the results, Section 6 discusses the results and the API implementation and finally, Section 7 describes the conclusions and future work.

2. Related work

This paper's contribution is related to work in the area of data exploration and supervised classification based on machine learning in the medical texts.

Previous work has been done in the overlapping fields of medical data mining, medical NLP or BioNLP and medical text mining (Chapman & Cohen, 2009; Van Der Spoel, Van Keulen, & Amrit, 2012). Relevant to this research are studies that focus on data mining or text mining in the (semi-) medical context. Closest to this research are applications of predictive analytics using unstructured (semi-) medical text.

2.1. Data and text mining in the medical context

Bellazzi and Zupan (2008) provide an overview of data mining in clinical medicine and propose a framework for coping with any problems of constructing, assessing and exploiting models. The emphasis is mainly on data mining in general, but useful guidelines are provided. Yoo et al. (2011) provided a similar literature review more recently. However, apart from research by Rao, Maiden, Carterette, and Ehrenthal (2012) towards the classification of breast- or bottle-feeding from unstructured data, no previous research uses unstructured semi-medical data for predictive analysis. Even more, there is no precedent of identifying child abuse from semi-medical texts.

Many text mining studies in the medical context are focused on extracting structured knowledge from medical text or notes. Efforts towards creating a pipeline for analysing medical notes include the work by Goryachev, Sordo, and Zeng (2006), who define three typical uses of such pipeline: to match concepts, to construct a classification model, or to automatically encode the documents using for example Unified Medical Language System (UMLS) or Medical Language Extraction and Encoding System (MedLEE). Zeng et al. (2006) then use the pipeline to extract data related to asthma from free text medical notes. Automatic encoding makes up a significant part of the available literature, including work by Friedman, Shagina, Lussier, and Hripcsak (2004), Zhou, Han, Chankai, Prestrud, and Brooks (2006) and Hyun, Johnson, and Bakken (2009). Other efforts include extraction of disease status from clinical discharge texts by Yang, Spasic, Keane, and Nenadic (2009). An overview of such research with the aim of supporting clinical decisions is given by Demner-Fushman, Chapman, and McDonald (2009).

Most of the developed tools and methods involve the English language, but there are a few occasions in which Dutch medical language was studied. Spyns and De Moor (1996) pioneered with a Dutch Medical Language Processor (DMLP) focusing on the language-specific parts of the language processing chain or pipeline. They later evaluated their work with four applications, concluding that although work still had to be done, the results were very promising (Spyns & De Moor, 1998). Up to that point, an overview by Spyns (1996) shows that only one attempt at processing Dutch clinical language had been done with the Ménélas project (Zweigenbaum, 1995). This project was mainly a free-text encoding effort.

More recently, Cornet, Van Eldik, and De Keizer (2012) provide an overview of the three tools available for Dutch clinical language processing with the goal of outputting Systematised Nomenclature of Medicine Clinical Terms (SNOMED CT) data. SNOMED CT is the most comprehensive collection of systematically organized medical terms in the world. It is multilingual and is used mainly to effectively record and encode clinical data. Cornet et al. (2012) also emphasize that a lot of research is needed towards a spelling checker, a negation detector and, importantly, a domain-specific acronym/abbreviation list as well as a concept mapper for the Dutch medical language. In summary, after the research by Spyns (1996) and Spyns and De Moor (1996, 1998) not much progress has been made over the last 20 years for Dutch medical language processing.

While this research is focused on machine learning thus automatic pattern extraction, it is interesting to review literature for indicators for abuse, which in contrast to obesity, is harder to capture in structured data. These insights can, later on, be used for selection of structured data. Structured data can be used to improve the model, and to test whether analyzing the unstructured data is even needed next to the structured data.

2.2. Child abuse

A comprehensive Delphi study[2] by Powell (2003) on early indicators of child abuse and neglect describes 5 physical (e.g. patterns of injuries), 13 behavioral/developmental indicators (e.g. self-harm, undue fear of adults) and 16 parenting (e.g. inability to meet basic needs of child, use of excessive punishment) indicators, that may occur separately or cluster together. Although the indicators of child abuse are an ambiguous topic, this Delphi study among 170 experts from different backgrounds does provide an interesting overview. The author herself points at a potential flaw of the study, as most of the experts on the panel held senior positions in

[2] A Delphi study is a research method that relies on a panel of experts. In several rounds, experts are asked for their opinion on a subject of disagreement. After each round, an anonymous summary of all experts judgments is provided and experts are encouraged to adjust their opinion in the light of this summary. Iteratively, consensus is to be achieved.

their organizations and did not reflect the first-line child protection. Moreover, in a generally favorable commentary of the review (Sidebotham, 2003), it has been stated that whichever (clusters of) indicators are used to alert people to possible maltreatment, they are not diagnostic and definitive proof of maltreatment, and that taking the step from a possible indication to a diagnosis of maltreatment requires clinical acumen and a holistic approach. A study towards the demographics of abuse was conducted by Jones and McCurdy (1992) identifying links with types of maltreatment. Unfortunately, the test group in their study consists solely of abused children and thus no relevant indicators of abuse were identified.

With regard to analyzing unstructured textual data in relation to (child) abuse, publications are scarce. Schrading (2015) extracted text datasets from Twitter and Reddit and analysed them using different natural language processing and machine learning techniques. The analysis revealed the reasons for leaving or staying in an abusive domestic relationship, while it also identified the stakeholders and actions in these relationships. Using a different methodological approach (a qualitative phenomenological study), Stacker (2016) explored subtle and non-regulated aspects of the forensic interviewing process of interviewing children that alleged abuse or neglect. The study provided a rich description of each forensic interviewers' experiences of the interview process while gathering information from alleged child victims. Truthfulness and informativeness of data gathered during interviews from children were stressed as primary goals by the study participants.

2.3. Predicting child abuse

More than 30 years ago the development of expert systems has been proposed for providing decision support to professionals in child protection agencies (Schoech, Jennings, Schkade, & Hooper-Russell, 1985). This is becoming a reality in today's environment, where a substantial amount of data from multiple sources is available on children and their families (Gillingham, 2015). Building upon the view of Schoech (2010), in an ideal scenario the gathered information should not be modified by the use of an expert system, while at the same time allowing potential modification of the existing patterns of information flow to be more efficient. Moreover, for child welfare workers this would mean good interoperability in a top-down model, where the expert system could assist with all manner of tasks, ranging from routine/inconsequential (e.g. recording information) up to those of critical importance, providing support with an assessment of risks pointing towards child maltreatment.

Predictive risk modeling (PRM) tools coupled with data mining and machine-learning algorithms should be capable of directing early interventions to prevent child maltreatment from occurring (Gillingham, 2015). Successful early intervention programs already exist e.g. the Early Start program in New Zealand (Fergusson, Grant, Horwood, & Ridder, 2006), however, there is a range of challenges that need to be addressed before coupling these programs with PRM. In addition to selecting reliable and valid outcome variables, while ensuring the consistency of their registration (Gillingham, 2015), there are moral and ethical challenges that need to be taken into consideration (Keddell, 2014).

Vaithianathan, Maloney, Putnam-Hornstein, and Jiang (2013) explored the potential use of administrative data for targeting prevention and early intervention services to children and families. Their data set was derived from public benefit and child protection records from the 57,986 children born in New Zealand between January 2003 and June 2006 and recorded until 2012. The final predictive risk model, with an area under ROC curve of 76%, included 132 variables. From the top 10% children at risk, 47.8% had been substantiated for maltreatment by age 5 years. Of all children substantiated for maltreatment by age 5 years, 83% had been enrolled in the public benefits system before the age of 2.

Horikawa et al. (2016) developed a linear prediction model (45.2% sensitivity, 82.4% specificity) using administrative data from 716 child maltreatment incident cases (stringently selected from 4201 cases reported to Shiga Central Child Guidance System, Japan) to identify the first recurrence of child abuse within the first year of the initial report. They identified and used 6 factors in their multivariate logistic regression model, namely the age of the child, the age of the offender, the history of abuse of the offender, household financial instability/poverty, the absence of guardianship and referral source.

Concluding on a harsher note, programmatic and ethical considerations were discussed by Church and Fairchild (2017), in an aftermath of the 2016 "Rethinking Foster Care" symposium, looking at the appropriate role of predictive analytics in child welfare. Church and Fairchild (2017) stated that while predictive analytics might improve the U.S. foster care system, child welfare agencies should insist on the transparency of algorithms, whenever they are used to identify at-risk children.

This study improves on previous research in many ways. It is based on a large dataset that is complete both in terms of quantity (all the children of the Amsterdam region over a 4 year time period) and quality (detailed information about every child included). Uniquely, this study takes both structured and unstructured data into account in building a prediction model. Moreover, insight is provided into the inner workings of the model (please see Section 5.13). Consequently, owing to the model's good performance it has been deemed useful for day to day operations, and we describe its implementation in a decision support system API for identifying child abuse in this study (see Section 5.15).

3. Models and methodology

For this research, data was provided from the child health department (JGZ) of the largest public health organization in the Netherlands, the GGD Amsterdam. In addition, JGZ also provided knowledge and expertise in the form of pediatricians in a scrum group. The data consisted of (partly medical) files on 13,170 children born in 2010 in the Amsterdam region, all reaching the age of four in 2015, at the time of this research. With on average 148 contacts with the JGZ per child, these visits resulted in 195.188 individual data entries. Of the 13,170 children, 657 children had been labeled presumably abused by the JGZ over the course of four years. It is important to note that the JGZ estimated that these 657 children account for 25%–30% of children that should have been labeled. An overview of the data's layout is given in Appendix B.

3.1. Data exploration

Table 1
Characteristics of the GGD data set.

Characteristic	Value
Number of children	13,170
Consults	195,188
Average number of consults per child	14.82
Average number of words per consult	41.58
Lexical diversity (nr of unique words vs. total nr of words) 1k random consults	0.16
Lexical diversity 1k random consults excluding stopwords	0.23

Quantitative characteristics of the data set are summarised in Table 1. Taking all children born in one year ensures relative randomness of the sample. The year 2010 was chosen, as the current information system Kidos and JGZ way of working were already in place and established, providing a stable environment for data retrieval, not needing additional data transforming steps between systems. With regard to privacy, any structured information that could be used to identify the child was removed, e.g. a unique identifier per child added by the JGZ to enable tying pieces of data together for one child. References to staff were handled in a similar manner. As described by Cios and William Moore (2002), this process of *de-identification* ensures anonymity but allows for the JGZ to trace back specific results to specific children.

3.2. Unstructured data

The data used in this research are the (semi-)medical notes written down by pediatricians or nurses into four, subject specific fields for note-taking per consult, the most voluminous being the *conclusion* field. This field contains a summary of the child and is hereafter referred to as SOC: summary of child. Some of the text is about the social dynamic of the family, describing the current situation, wishes of the parents and a number of medical diagnostics. The text contains numerous acronyms depending on the author and the team the author is part of. The average amount of words per consult is 41.58.

An example of a short note taken 4 months after the birth of another child is:

> *prima kind, m chron bronchitis advies begin fruit pas met 5 mnd*

Translating to English as follows:

> *Nice child, mother has chronic bronchitis, advised to not start with fruit until 5 months of age.*

3.3. Structured data and labels

In addition to the unstructured notes, we added specific structured data and labels to our data set that we used as features, and as dependent variables in our predictive model. These data/labels are i) "Findings ZSL", ii) "Action ZSL", iii) "Attention", iv) "Family relations" and v) BMI.

"Findings ZSL" represent worries in the social environment (Dutch - "Zorgen Sociaal Leefmilieu"), set to 1 if there has been a presumption of child abuse and 0 otherwise. In the analyzed data set, 628 out of 13,170 distinct children had this label set to 1. When we found that 'Findings ZSL' was wrongly set to 0, although nature of abuse was known, a corrective action was taken to include those children as well, amounting to 657 children additionally labeled as presumably abused. JGZ indicates that professionals should always set this to 1 for children that they presume to be abused, making "Findings ZSL" in principle useful as a dependent variable for a predictive model for child abuse. In reality, this happens only in about 25%–30% of the cases, either to prevent the risk of drawing a wrong conclusion and hurting the bond of trust with the parents or because the health professional takes action without registering it. This leads to noise in the data, to incorrect management information and more concretely to missing abused children in the data.

The "Attention" label is set to 1, if, due to any reason, (extra) attention needs to be paid to this child, with no clear directive given. 2459 out of 13,170 children, approximately one in five, have this label set to 1.

"Family relations" are summarized in a table, containing relation types (e.g. brother, mother, adoptive father etc.) and ID (birth date only) of the relative.

Also, when a child visits for a consultation, he or she is measured and weighted, resulting in tables of lengths, weights and Body Mass Indexes (BMI).

4. Pre-processing

4.1. Storage

Before the data can be processed, we inserted the data into a MySQL[3] database, allowing for easy filtering and drilling down on dimensions. With the prospect of engineering features per child, there are multiple fact tables containing, for example, the *summaries of child* (SOC). The data for distinct children are stored in a dimension table. This way, features like the number of consults or whether or not a child is obese can be extracted to one flat feature table. Other fact tables include the Body Mass Indices (BMI), ZSL and Attentions linked to the children. Other dimension tables include the action types, the locations of the consults and the practicing pediatricians.

4.2. Terminology normalization

There are many abbreviations and acronyms used in the texts that enabled quick input of data, most of these were imposed standards among the JGZ personnel. Acronyms such as *P* for *papa* translating to 'father' in English and *ZH* for *ziekenhuis* translating to 'hospital' in English. We used regular expressions (RE) to extract all short abbreviation-like words can be extracted from the data, e.g. words consisting of less than 4 characters that are all consonants and possibly contain dots. This is in agreement with the method utilized by Xu, Stetson, and Friedman (2007). The extracted acronyms and abbreviations are then ordered by frequency of appearance and enriched with a sentence in which the acronym appears using the NLTK *concordance* function.

Next, we asked the subject experts, in this case the medical staff of the JGZ, to explain the list of acronyms. We then converted the terms to regular expressions and formed tuples with their respective replacement. We used a Python script to loop through the data for RE-based string replacements.

4.3. Trivial word removal

We removed all indications of time for two reasons; they did not contribute to the identification of abuse and they varied constantly introducing noise in the data. Similarly, dates and times were both removed using RE. Finally, all left-over numbers or words containing numbers were removed.

Another common preprocessing step is the removal of stop words; words that appear very frequently but do not attribute to the meaning of the text. For this, we used a standard Dutch language stop word list, which is included in the Natural Language Toolkit (NLTK) package. Amongst the stopwords were a number of negative words like "niet" and "geen", or not and no/none in English, that were not removed from the text. This was done to make sure that when n-grams with $n > 1$ are used, the meaning of the text *"not good"* is captured.

4.4. Stemming

Words in the SOCs appear in many forms and tenses, whilst pointing to the same concepts. These various forms lead to a more flat distribution of word quantity: more unique words and less volume for the top terms. This is not beneficial for classification algorithms that need to identify common terms and themes. In order to group various forms of the same words together, all words were reduced to their stemmed form using the Dutch Snowball stem-

[3] http://www.mysql.org.

mer[4]. This stemming framework proposed by Porter (1980) is included in the Python NLTK package.

4.5. Tokenization

Depending on the method of classification, the texts needed to be split up into sentences or just sequences of words. Although splitting sentences seems like an easy task, it is very hard to perform algorithmically. A sentence might end with any of '.!?' followed by whitespace and a (capital) character, but not only do quick notes often not comply with this rule, also this combination of characters is frequently seen in abbreviations or medical terminology. We, therefore, used *tokenization* to split the text into words. This also allowed us to use n-grams, combinations of *n* sequential words, in the analysis later on. Because all noise regarding line-breaks, special characters and white-space had been removed with RE, tokenization was easily done by splitting the text using the single white-space character.

4.6. Extraction of possible textual features

In order to explore possible features in the SOCs for predictive analysis, we used *force* specific clusters and found the most common words or combinations of words. This was done for the whole body of texts and for specific groups of children. In this way, we could uncover words that were features for identifying groups of children, and could this could help us identify the groups that suffer from abuse. We visualised the words and their relative frequency using word clouds and discussed the results with the JGZ, to demonstrate the distinctiveness of groups within the population of children. These resulting word clouds showed some obvious and some interesting terms as being distinctive for a group. This indicated that the topics that should be predicted were described in the SOCs, and thus the SOCs could be usable for further text mining.

4.7. Extraction of possible summarising features

For free text data, features can be extracted from the content of the data as well as from the form of the data. The latter are called summarising features and can be very relevant for the performance of the classification model. To illustrate this, we explored two summarising features for child abuse.

SOC length as a predictor of presumed abuse. A typical feature that summarises free text is the length of the text. The reasoning that leads to this feature is that more extensive documentation could be made for children that have some health issues (like abuse). We tested if the length of the text (SOC length) could be used as a predictor for presumed abuse. The groups of children were split using the enhanced ZSL finding column calculated per age interval of 0–1, 1–2, 2–3, 3–4. These age intervals were used because we wanted to test whether the SOC length differs significantly between the groups, and from what age could the difference be significant. Fig. 1 indicates how distinctive the length of the consult really is.

The p-values in Table 2 show that the difference in SOC length between the presumably abused children and the other children is significant. Therefore, the average length of the SOC can be used as a feature in a predictive model for presumed abuse. It makes sense that the difference between mean values rises over the years, for at the end of year 4, all children that are presumably been abused between 0 and 4 have the label *ZSL finding*. In contrast, between the ages 0 and 1 we would expect about 25% of children that end up with a *ZSL finding*, already have such a label.

Table 2
Mann–Whitney *U*-test *p*-values for average SOC length per age interval for presumed abuse.

Interval	Test statistic	*p*-value
0–1	2262729.0	1.50948e−45
1–2	2253800.0	1.26391e−34
2–3	1843716.0	4.18773e−39
3–4	1888747.5	3.78134e−39

Table 3
Mann–Whitney *U*-test *p*-values for consult quantity per age interval for presumed abuse.

Interval	Test statistic	*p*-value
0–1	2575126.5	2.67548e−25
1–2	2110410.5	2.76681e−45
2–3	1517366.0	2.48735e−71
3–4	1871244.5	1.22809e−40

Consult quantity as a predictor of presumed abuse. The number of consults can also possibly be used as a feature in a predictive model for presumed abuse. We used the same groups as previously, using the *ZSL finding* variable. The resulting box plot indicating a difference in the average number of consults can be seen in Fig. 2. Again the *p*-values in Table 3 indicate that the differences are significant and that this summarising feature can be used as a predictor for presumed abuse.

Other summarising features include the lexical diversity of the SOCs, the average time span between consults or the number of distinct medical professionals a child has come in contact with.

From the previous sections, it is evident that the data contains textual and summarising features that could be relevant in the creation of a prediction model for presumed child abuse. One of the contributions of this research is the use of unstructured JGZ data in predicting presumed child abuse, but the main goal of this research was to have a usable prediction model. The JGZ had little structured data available that they had linked to suspicion of abuse. In the next section, we describe our model using structured data and compare it with the one build using unstructured data.

4.8. Data sampling

With just 5% of the data belonging to the positive group, the data was relatively unbalanced. To be able to use the data for classification modeling, we used the *random under-sampling* (He & Garcia, 2009) approach. To ensure a large enough training set to cover the various types of abuse, our training data consisted of half of the positive group and an equal number of files from the negative group. We sampled both at random from the positive and negative groups respectively. k-fold cross validation is usually limited to two folds when taking half of the positive group for training. By repetitively sampling the data randomly, cross-validation was possible with more than 2-folds.

4.8.1. Term weighting

We tested several weighting schemes for possible improvement of the classifier performance: Boolean occurrence, count, tf-idf augmented for varying text lengths (Manning, Raghavan, & Schtze, 2008), DeltaTF-IDF (Martineau & Finin, 2009) and BM25 (Robertson, Walker, Beaulieu, & Willett, 1999).

[4] http://snowball.tartarus.org

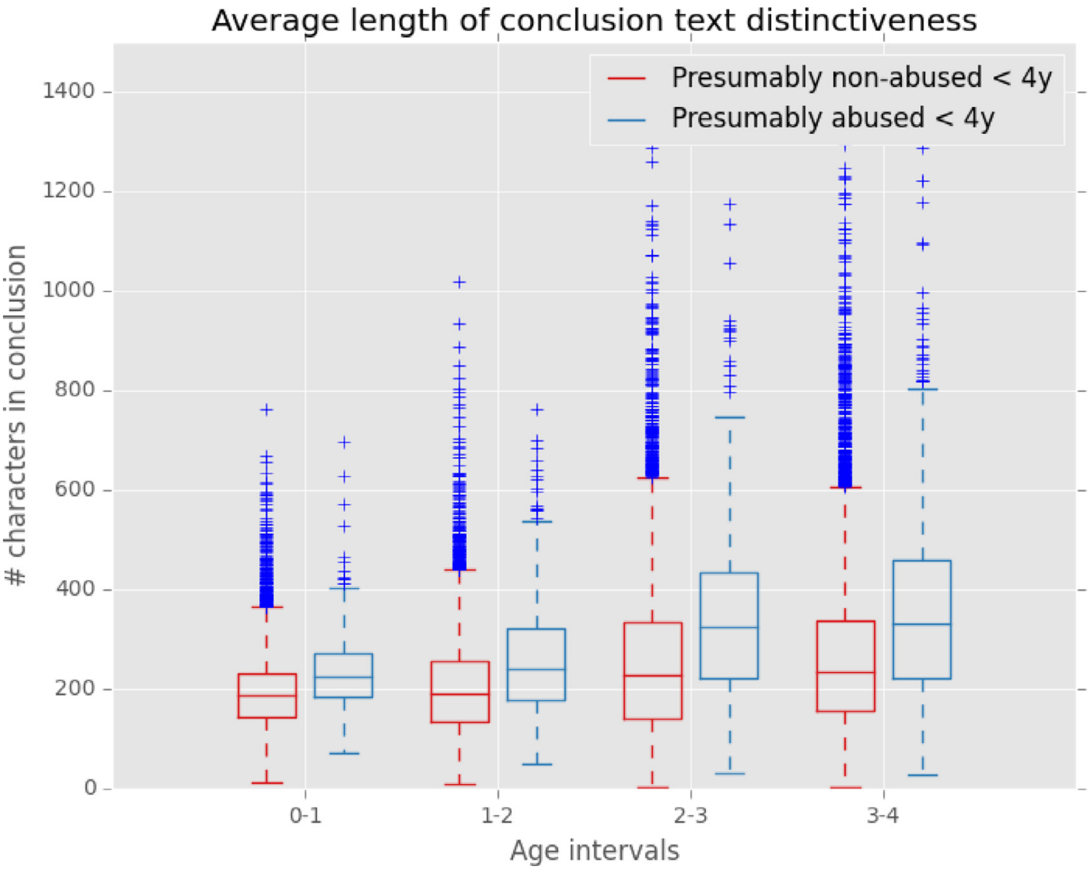


Fig. 1. Box plot of the average SOC lengths per age interval for presumed abuse.

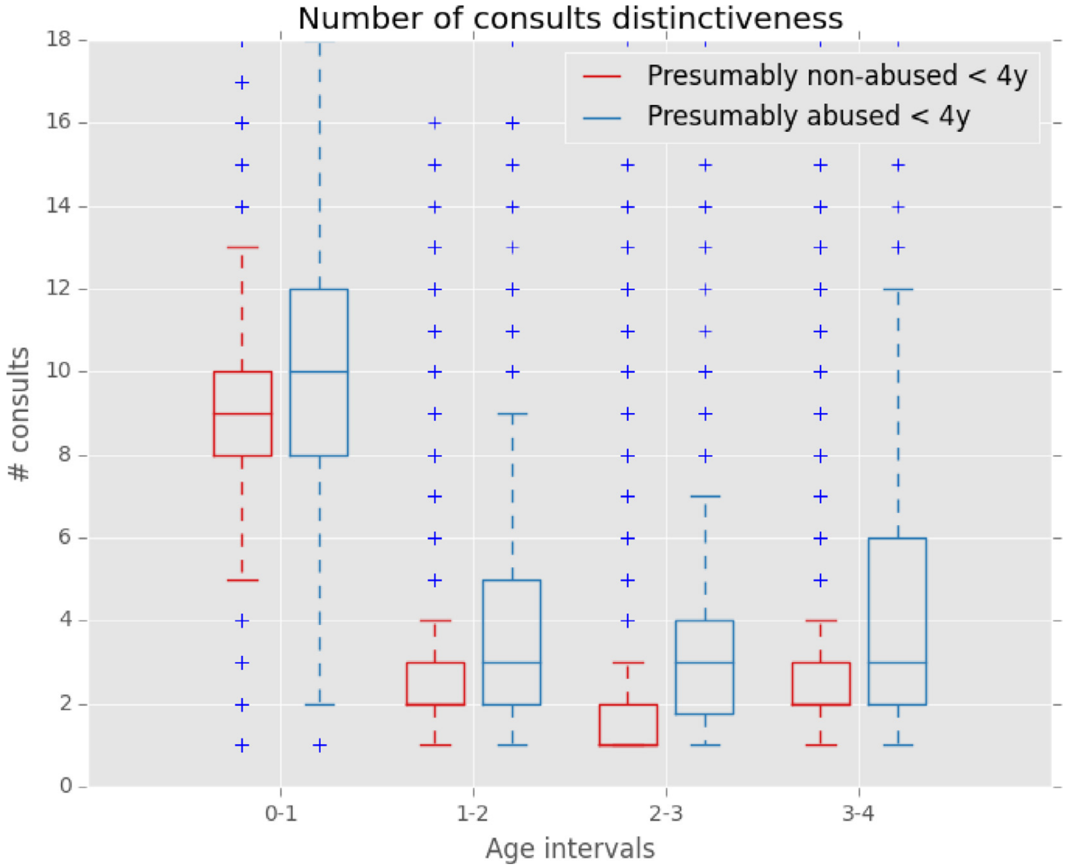


Fig. 2. Box plot of the consult quantity per age interval for presumed abuse.

4.9. Classification models

We used the three most popular algorithms (Aggarwal & Zhai, 2012) for this classification task: Naive Bayes (Kononenko, 1993), Random Forest (Ho, 1995) and Support Vector Machine (Drucker, Wu, & Vapnik, 1999). Although non-linear algorithms like Neural Networks have received much attention lately, we did not consider them, due to their computational heaviness and history of not consistently outperforming less sophisticated methods in text mining (Aggarwal & Zhai, 2012). We employed the Python implementations of the algorithms, as provided in the widely used Scikit-learn package (Pedregosa et al., 2011). We used the same algorithms to classify the structured and unstructured data because of their flexibility and ability to cope with a sparse, high-dimensional feature space (Aggarwal & Zhai, 2012), that is typical for text mining.

5. Evaluation

After the exploratory analysis, we endeavor to predict whether a child suffers from abuse using classification models.

5.1. Performance metrics

It is important to consider the performance metrics before modelling, for these metrics dictate when a model is performing well. Typically there are many trade-offs between these metrics when optimising a model: improving the model on one metric will decrease the score on another.

In our model, True Positive (TP) implies a correct classification of an abuse presumption. False Positive (FP) are children that are classified as presumably abused but are not labelled as such by the JGZ. False Negative (FN) are children that have been labelled by the JGZ but are not classified as such by the algorithm. Lastly, True Negative (TN) are correctly classified children that have not been labelled presumably abused.

The model will be used to find children with a condition, so focusing on recall is important, for as few FN as possible should be predicted. The balance between these two goals is captured in the Receiver Operating Characteristic (ROC) curve and its summarising metric Area-Under-Curve (AUC). Optimising the ROC curve and AUC will be the directive for optimisation of the model. To prevent the classification model from assigning every child a presumably mistreated model, accuracy and balanced accuracy should also be taken into account, even more so since the classes are very unbalanced ($p' = 657$ versus $n' = 13{,}137$).

As we described in Section 3.3, there was a lot of noise in the dependent variable *ZSL finding*, that is our dependent variable of our model. The JGZ indicates that for every child who has a correct ZSL label, around 3–4 children would have an incorrect label. It is, therefore, unclear if patterns indicating abuse will be found not only in the group with the label ZSL but even more so in the in the non-ZSL group. As a result, the amount of FP will always be high and the precision low. Indeed, if the precision would be optimised too much by lowering the amount of FP, the goal of spotting children that might be abused is not achieved. We can, therefore, expect the precision to be low due to the noise in the data, which is precisely the problem that we aim to solve by deploying this model for decision support. Thus, though we mention precision scores in the optimisation tests they are not so critical in determining the quality of our model.

5.2. Data selection

We assigned the value 1 for presumably mistreated (ZSL) if at any point the particular child had been labelled so by JGZ, and 0 if not.

Table 4
Setup of benchmark classifier for ZSL.

Setting	Value
Input data	Processed SOCs for age interval 0–4
Features	Count for 100 most common words
Classifier	Random Forest (n estimators = 100)

Table 5
Benchmark performance scores for ZSL classification.

Metric	Value
Precision	0.1617
Accuracy	0.8050
Balanced accuracy	0.7949
Recall	0.7837
F1	0.2679
AUC	0.8738

Initial experiments proved that a training set of $n'_{train} \cup p'_{train}$ containing a ratio of n'_{train} versus p'_{train} similar to n' versus p' in the entire data set, that is 20 non-abused children to one presumably abused child, led to very inferior classifier performance. Therefore, the training set was made up of a randomly selected 50% n'_{train} and 50% p'_{train}, an equal amount of presumably abused and non-abused children. To ensure a large enough training set whilst retaining enough data for testing, a training set containing $n'_{train} = 325$ and $p'_{train} = 325$ was initially used. Thus, half of the presumably abused data was used for training. Taking half of the total positive population for training makes it hard to do cross-validation of the results with more than 2 folds, so both the 325 n'_{train} and the 325 p'_{train} were chosen at random from n' and p'. This left us with a test set of $n'_{test} \cup p'_{test}$ with $n'_{test} = n' - n'_{train}$ and $p'_{test} = p' - p'_{train}$ for each iteration of cross-validation. We typically used 10-fold cross-validation, each time randomly sampling half of p' and n', so the chances of a sample left unused are very small. In order to maintain the ratio between the classes in the test set as it is present in the whole data set, we used $n'_{test} = 0.5 \times (n' - n'_{train})$. Thus, we had 20 non-mistreated children for every presumably abused child in the test set. We did this as it is important to have the same ratio of p versus n in the test data as in the whole data set, to make sure that the performance metrics are representative for a real life application of the model. Fig. 3 provides a visual overview of this sampling method, described as random under-sampling (He & Garcia, 2009).

Every individual model created is indeed based on just half of the randomly selected positive samples in the data set. For feature building, model selection and parameter tuning this would suffice, but for taking a model into production, it should contain as much knowledge as possible. Therefore, when creating the final model for implementation, we used the entire set of positive samples for training, and a test set from a different birth year for validation.

5.3. Benchmark performance

We first create a benchmark to compare our models with. For this, we use a standard approach to text classification with properties as described in Table 4.

Table 5 shows the mean performance scores for the model on the test set, using 10-fold cross validation.

The algorithm scores quite well on accuracy and recall but has a low precision score. With the classes in the test data being imbalanced, this implies the presence of many FP instances. This is confirmed by a typical confusion matrix from the test, which in vector form is (TP = 276, FP = 1439, TN = 5529, FN = 56), where

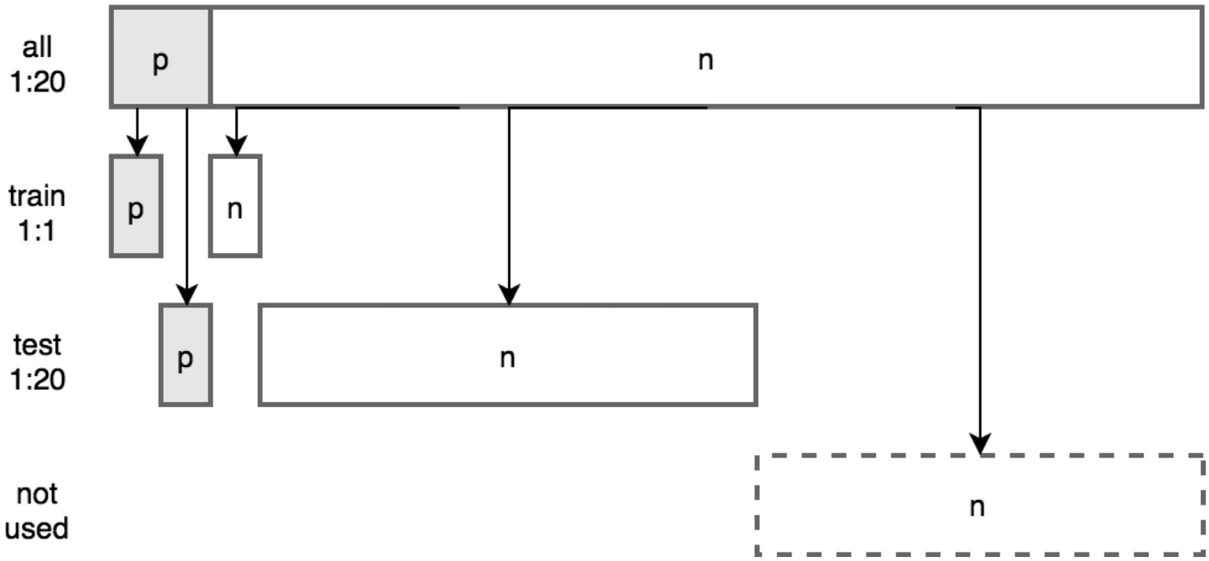


Fig. 3. Schematic view of data sampling for a single training run.

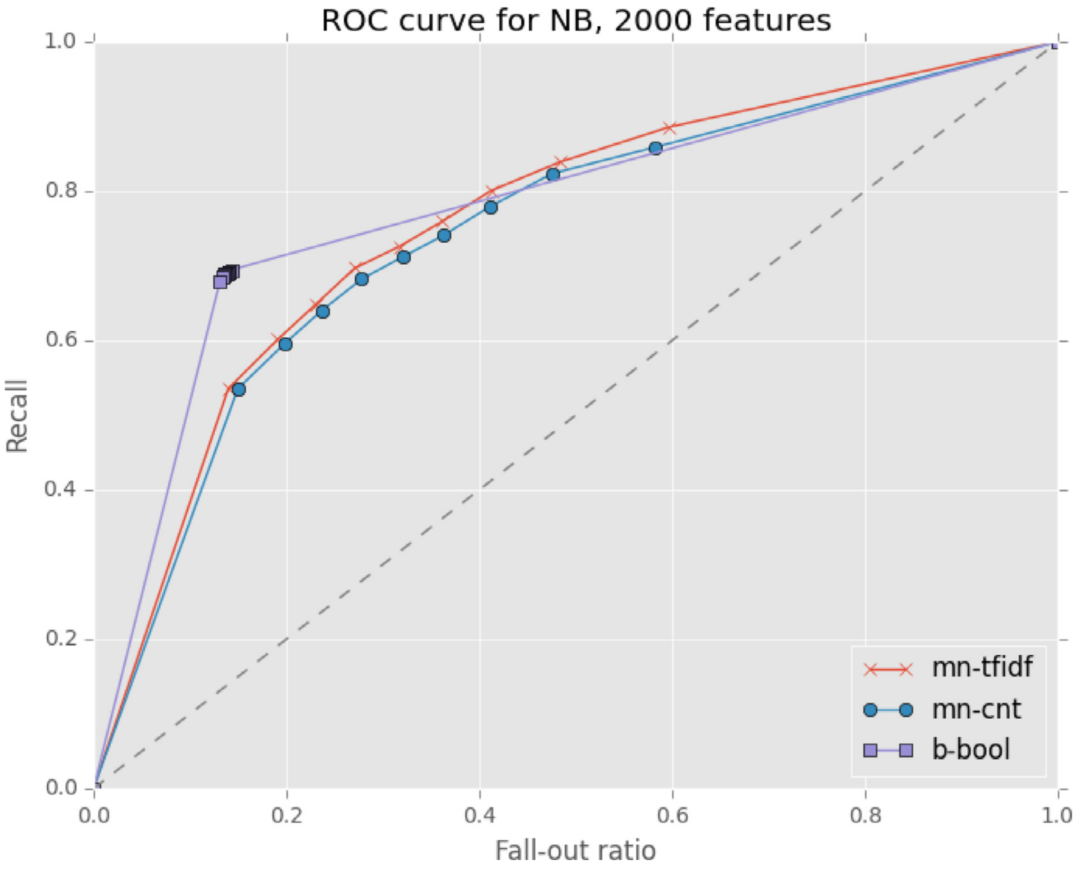


Fig. 4. ROC curves for Naive Bayes algorithms.

for every TP there are about 5 FPs. Nonetheless, we obtain an AUC of 0.87, that already shows promise. (Fig. 4)

5.4. Feature building

The most important features for this bag-of-words approach are the occurrences of words in the texts. A common way to decrease the dimensions of the feature vector is to apply univariate statistical tests to select a top number of features, like the ANOVA or χ^2 test. In Section 5.5 these tests are applied with various parameters.

In addition to the words as features, we derived some other features from the data given by the JGZ. We appended these to the word-features and tested them for relevance. The following features passed our statistical test:

- Average amount of characters per consult
- The most frequently visited JGZ location

Table 6
Performance of NB algorithms.

	Accuracy			Recall		
Features	mn-tf-idf	mn-cnt	b-bool	mn-tf-idf	mn-cnt	b-bool
100	0.7099	0.8878	0.7532	0.5993	0.6511	0.7277
500	0.7406	0.8929	0.8392	0.6262	0.6851	0.6652
1000	0.7461	0.9023	0.8452	0.6270	0.6546	0.6844
2000	0.7272	0.9094	0.8697	0.6837	0.6553	0.6511
5000	0.5629	0.9186	0.8913	0.8170	0.6142	0.6149
10000	0.3029	0.9158	0.8998	0.9376	0.6184	0.6355

While the following features proved not to be relevant include:

- Lexical diversity
- Count of family relations per type
- Gender

Categorical features like the most frequently visited JGZ location were dummified.

5.5. Algorithm tuning

In this section, we describe how we tuned, analysed and compared the algorithms. There are two ways to approach this problem: using classification algorithms, or by using anomaly detection algorithms. The reason for using anomaly detection is the imbalanced data: when the minority class makes up a very small percentage of the total data we can effectively do anomaly detection on the majority class. We first apply several classification algorithms, and use a one class SVM to test the anomaly detection approach.

5.6. Naive Bayes for classification

The Naive Bayes model can be applied in two forms: Bernoulli and multinomial. The input of the multinomial algorithm can be weighted, which is often done using tf-idf weighting. These can be smoothed using Laplace or Lidstone smoothing, to account for features that are found in the test set but not in the training data. In this case, especially when using a feature weighting scheme, smoothing will probably not do much for performance. Table 6 contains the results of the grid search that is used to approximate optimal configuration of the Naive Bayes algorithms. The configurations are coded using *mn* for multinomial, *b* for bernoulli and the feature weights *tf-idf*, *cnt* for count and *bool* for boolean.

The outcomes indicate that there is a strong trade-off between the accuracy and recall for the Naive Bayes classifier, and *tf-idf* does not seem to improve the performance. It is unclear whether the multinomial classifiers outperform the Bernoulli variant, but looking at the ROC curves for all three at 2000 features, it is clear that the "simplest" Bernoulli classifier with boolean input performs the best. Though the precision for this classifier is 0.25, the Bernoulli classifier with Boolean features performs best at an AUC of 0.779. On further testing, we found that we reached this score from 50 features and it remained constant up until 10,000 features. We also found that pre-selection of features using χ^2 had the most impact, increasing the AUC to 0.817.

5.7. Random forest for classification

We applied a Random Forest (RF) for our decision tree algorithm instead of utilizing a simple decision tree, as it is known to significantly improve the model's performance. Tuning the performance of a Random Forest mainly comes down to selecting the features that are used, the number of trees, and which splitting criterion to employ. We selected the features using ANOVA (shown as *an*) and χ^2 tests of relevance, as a RF does not cope well with a very large number of features. For the number of trees, we chose it to be equal to the number of features, as is usually done. For the splitting criterion, we used the default Gini Index splitting criterion in the Scikit-learn package. Table 7 contains the performance results for the models.

From the table, it can be easily seen that weighted features result in a higher performance than non-weighted, counted, features. Regarding accuracy, the ANOVA and χ^2 feature selection tests scores are similar. For recall, the χ^2 test seems to outperform ANOVA when the number of features becomes large. Increasing the number of features results in marginally better accuracy, while the best recall is found by using just 200 features for almost all versions of the algorithm. The precision for this classifier is 0.26. We use an ROC curve to compare the most promising algorithms: *an-tf-idf* and χ^2*-tf-idf*. A plot of these curves can be seen in Fig. 5.

It is clear that there is not much difference between ROC curves. The computed AUC for χ^2 selection with 200 features, 0.903, is slightly higher than the other algorithms at 0.899. The DeltaTF-IDF[5] variant of tf-idf leads to an AUC of 0.888, which is lower than the original tf-idf. The BM25[6] weighting scheme that according to Paltoglou and Thelwall (2010) performs best for binary text classification gives an AUC of 0.896. The implementation of these schemes in Python makes them computationally intensive and model run times are 5–10 times longer than the standard tf-idf scheme.

5.8. SVM for classification

SVM is known for the lack of many parameters that can be used for tuning as it does the tuning for a large part internally already. That leaves the user with the type of kernel to be used and the penalty parameter C of the slack error term to tune. We obtained reasonable results with either linear, polynomial or (Gaussian) radial basis function kernels. To approximate the best parameters to use we used a cross-validated grid search, repetitively training a model with a different combination of parameter values. Each model was then tested and cross-validated on the held-out test data, optimizing for the AUC metric. The relevant accuracy and recall metrics are shown in Table 8.

The precision of this algorithm ranges between 0.21 and 0.23. It can be seen furthermore that the highest F1-score and lowest fall-out rate are achieved when using the polynomial C=0.2 algorithm. As recall is an important metric for this model, the polynomial and linear algorithms are compared using a ROC curve for C in [0.2,0.5,1.0]. Fig. 6 shows the resulting curves.

The ROC curve shows that over the range of C-values, the linear kernel always outperforms the polynomial kernel. Within the linear kernel, we find that the differences are marginal and a C-value of 1 can be safely chosen. From here the optimal amount of *tf-idf* weighted textual features can be found at 1000 with an AUC of 0.906, although using up to 10,000 features can improve recall slightly at the cost of lower accuracy. Pre-selection of features using various methods like ANOVA or χ^2 does not effect the model's performance, which is explained by the fact that SVM already selects the most important features into the support vectors.

We also tested advanced versions of the tf-idf weighting scheme, but they did not result in an improvement of the performance. The DeltaTF-IDF scheme leads to an AUC of only 0.809 and for the BM25 we obtained an AUC of 0.884, both less than the basic tf-idf.

[5] https://github.com/paauw/python-deltatf-idf

[6] https://github.com/paauw/python-bm25

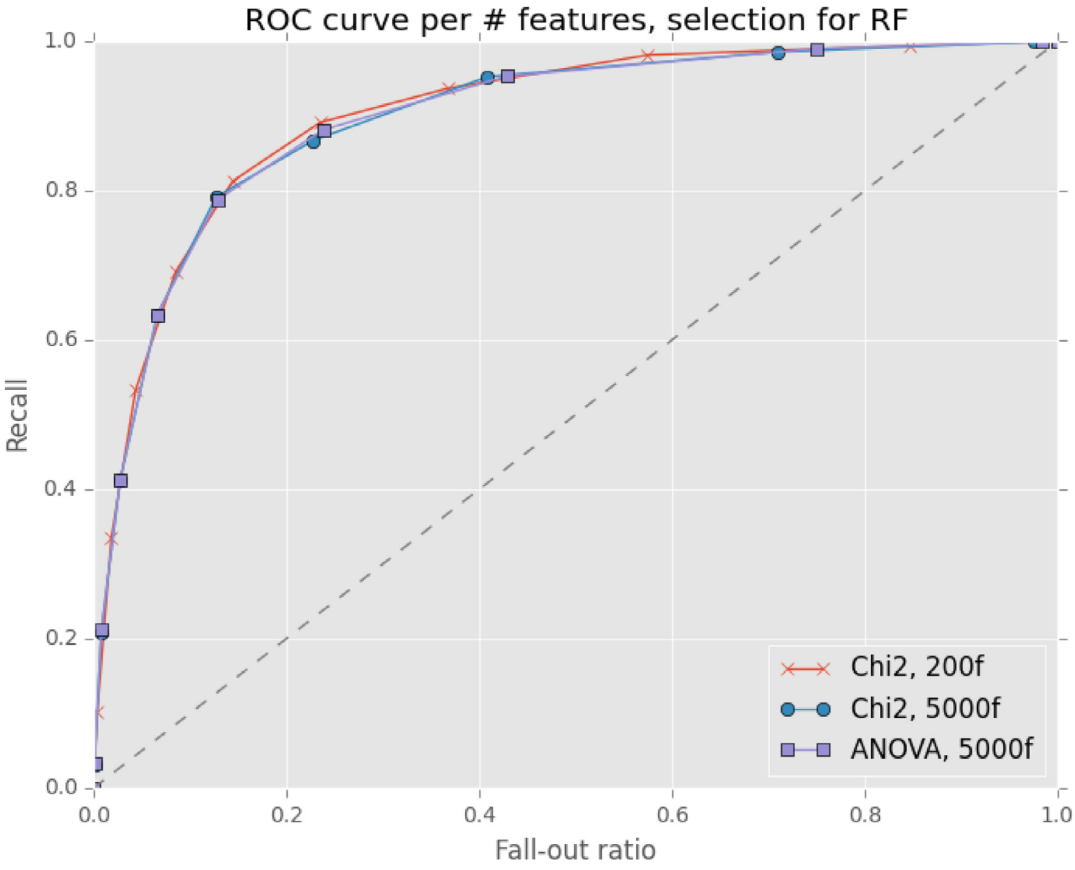


Fig. 5. ROC curves for Random Forest algorithms.

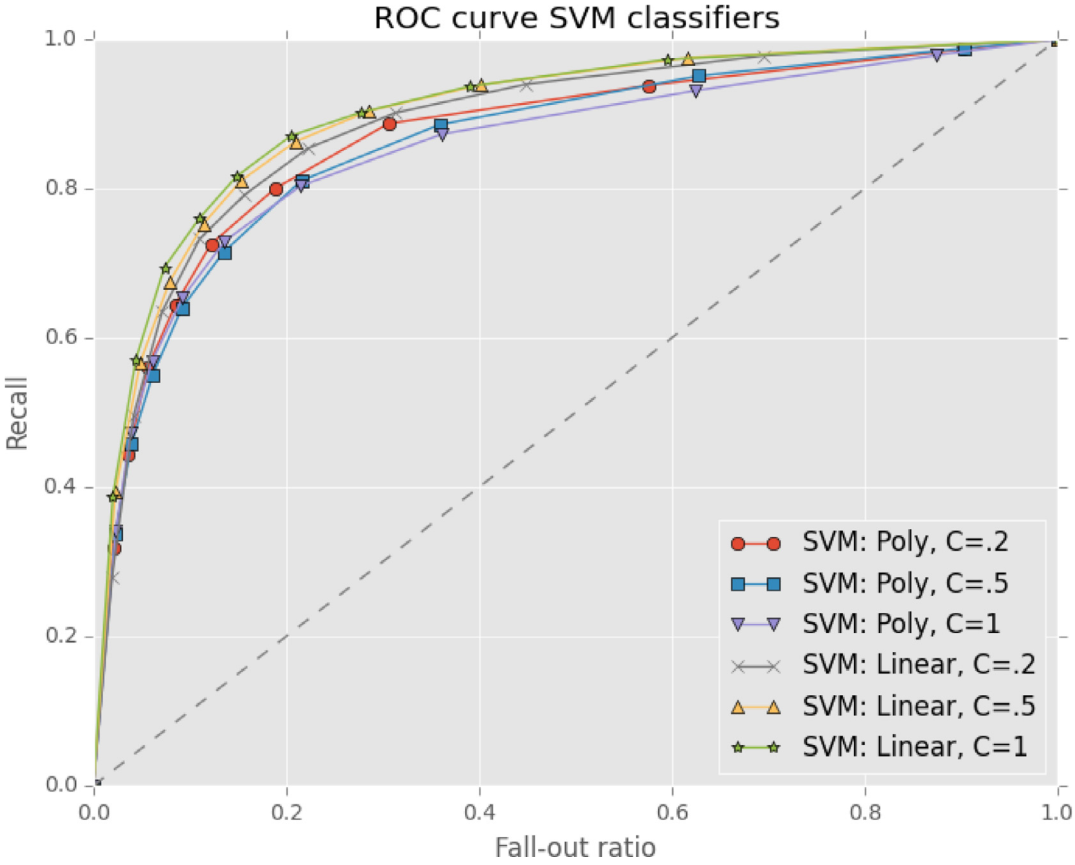


Fig. 6. ROC curves for linear and polynomial SVM algorithms.

Table 7
Performance of Random Forest algorithms.

	Accuracy				Recall			
Feat	an-cnt	an-tf-idf	χ^2-cnt	χ^2-tf-idf	an-cnt	an-tf-idf	χ^2-cnt	χ^2-tf-idf
100	0.8357	0.8496	0.8373	0.8566	0.7702	0.7965	0.7546	0.7865
200	0.8328	0.8507	0.8476	0.8603	0.7830	0.8121	0.7801	0.8177
500	0.8443	0.8651	0.8368	0.8638	0.7652	0.8092	0.7936	0.7993
1000	0.8434	0.8705	0.8441	0.8626	0.7709	0.7830	0.7723	0.7993
2000	0.8501	0.8698	0.8533	0.8669	0.7681	0.7823	0.7865	0.7908
5000	0.8521	0.8768	0.8489	0.8741	0.7816	0.7468	0.7731	0.7901

Table 8
Performance of SVM algorithms for classification.

	Accuracy			Recall		
C	Linear	Poly	RBF	Linear	Poly	RBF
0.2	0.8355	0.8801	0.7150	0.7929	0.7220	0.6348
0.5	0.8496	0.8596	0.7121	0.7972	0.7170	0.6383
0.8	0.8491	0.8389	0.7171	0.8121	0.7518	0.6319
1	0.8425	0.8461	0.7198	0.8248	0.7383	0.6461
5	0.8168	0.8128	0.7085	0.7731	0.7340	0.6702
10	0.7982	0.7947	0.6920	0.7766	0.7411	0.6915

Table 9
Performance of SVM algorithms for anomaly detection.

	Accuracy			Recall		
ν	Linear	Poly	RBF	Linear	Poly	RBF
0.048	0.8070	0.8104	0.8686	0.0068	0.0040	0.1200
0.143	0.8099	0.8127	0.8000	0.0028	0.0012	0.2877

5.9. Anomaly detection

As explained earlier, the problem of finding possibly abused children can also be approached as an anomaly detection problem. We applied a one class SVM algorithm and tuned it in a similar way as for classification, with an equal number of features. An important extra parameter is the ν parameter indicating the expected fraction of the data that is an anomaly. We tried two values for ν: the fraction of positives in the dataset, and the fraction of positives multiplied by 3 to account for the suspected under-registration by the JGZ mentioned before. The implementation of the algorithm in Scikit-learn automatically tunes the γ parameter.

Calculation of the performance metrics like the recall was again done for the minority class: the possibly abused children. Again, we used a grid search to approximate the right parameters as well as the kernel used for the one class SVM algorithm. Table 9 shows the results: the RBF kernel is the only feasible kernel to use, leading to a AUC score of 0.558 when using $\nu = 0.143$. As this is quite low compared to the AUC results from RF and SVM classifications above, we decided to not pursue anomaly detection further.

5.10. Additional structured data

From literature, Section 2.2, and the expertise of the JGZ's professional, several structured features were identified that might further enhance the model's performance. The JGZ does not have data on all indicators described in the literature, but demographics and a number of other relevant features are available. Table 10 contains the data made available by the liaison officer for child abuse. To see whether features built from this data were distinctive and contributed to the model, we applied a χ^2 test with respect to the presumed child abuse variable. Table 10 shows only the outcomes with $p - values < 0.05$.

Table 10
Structured data feature distinctiveness.

Feature	χ^2
JGZ location most visited*	multiple
Birth country of child and both parents*	multiple
Average characters per text	10373.144
Age of the mother at child birth	36.521
Special consultations on skin issues***	–
Dramatic event***	99.566
(Semi)permanent medical condition***	–
General health and disease***	33.080
Women's genitalia***	–
Mother's health***	54.365
Micturition / defecation***	6.126
"Samen Starten weging"***	296.235
"Triple-P"**	
1st contact	–
2nd contact	–
Burden vs. Carrying***	
Family	104.438
Child	18.050
Environment	45.623
Parents	90.844
Dental care***	5.493
Overweight***	9.465
General care received**	402.725
GGD care received**	111.810

*Categorical variable, **Count of occurrences, ***Count of occurrences with findings

Most classifiers can only deal with numeric features and some features are categorical, like the countries of birth. We converted these to sparse matrices with columns enumerating the categorical values. These are so many, that they have not been included in Table 10. Some birth countries correlate significantly with the presumed child abuse variable, but mainly due to the low volume per country.

For this research, there are two interesting applications of this structured data: testing whether a model based on structured data outperforms the model based on unstructured data, and then comparing it with a model build on both structured and unstructured data.

5.11. Unstructured versus structured data

To test whether it makes sense to use the unstructured data for prediction of presumed abuse instead of using the structured data, we built a classifier based solely on the structured data. We used all the variables stated in Table 10 with a Random Forest classifier, that resulted in the performance outcomes in Table 11. The model's performance is not as good as the model based on unstructured data with respect to recall and accuracy, but the AUC is only slightly worse. Note that the model based solely on structured data makes indirect use of the unstructured data, by using the average count of characters as a feature in this model. Removing this feature only marginally decreases performance, indicating that the

Table 11
Performance scores for structured classifier.

	Algorithm	Precision	Accuracy	Recall	AUC
Structured	RF	0.185	0.828	0.817	0.892
Combined	SVM	0.200	0.839	0.844	0.909
Ensemble	RF + RF	0.187	0.822	0.870	0.914

feature is mainly important when it can interact with textual features in text mining.

5.12. Structured and unstructured ensemble

While a model based solely on structured data did not challenge the model based on unstructured data, a model based on both might perform better than either one individually. In order to implement the model for the JGZ's daily operations, we need to make sure the model is as good as it can be. Either one model can be built incorporating both forms of data as features, or an ensemble method can be constructed of two models with balanced voting on the outcome.

The performance of the combined and ensemble classifiers is shown in Table 11. For the ensemble method, a Random Forest algorithm is used for the structured data part and both an SVM and RF were tested for the unstructured part. Since the SVM scored slightly worse (AUC of 0.911) than the RF, the latter was presented in the table. The voting is implemented by calculating the average chance of assignment to the positive class between the two classifiers. The outcomes here are the mean results of a 20-fold cross-validation. A 1-sided ANOVA test showed that, with $\alpha < 0.05$, the mean AUC's for the classifiers are not equal. We then used a two-tail t-Test to test if all means differ. Again with $\alpha < 0.05$ they do, but the difference in means between the combined method and the ensemble method is only just significant (with the t-Stat 0.01 point higher than the critical value).

5.13. "Opening the black box": sensitivity analysis

A limitation of using machine learning for decision support could be the limited understandability of the model for the end users. Especially in sensitive application areas such as the detection of child abuse, practitioners would like to know how the model works in specific cases. In this research, this limitation was partly mitigated by involving the end users from the start in the development of the model. To provide end users more insight into the inner workings of the text-based classifier we used the Lime project,[7] based on Ribeiro, Singh, and Guestrin (2016). This involves doing a local sensitivity analysis for a specific instance in the data. An example of this method is included in Appendix C along with further specific details.

5.14. Qualitative evaluation and user acceptance

One of the challenges of our implementation at the JGZ was having a prototype based on our prediction model accepted by the employees at JGZ. We, therefore, decided to get a small random sample of files and their respective model predictions reviewed by the liaison officer for child abuse at JGZ. This is not relevant for validation of the model, as the model already contains the judgments of around 500 professionals, and validating outcomes by one professional would not be statistically sound. Using such a predictive model in practice, however, is a big step towards acceptance of the results for a successful implementation. 29 children's files where selected at random from the group of false positives (FP) and false negatives (FN), 20 false positives and 9 false negatives. More FP files were selected, as the FP files were most interesting from the perspective of JGZ's work and time that could be spent by the liaison officer was limited.

The liaison officer for child abuse reviewed the files to see whether she agreed with the model. She came to the following conclusions:

- In 13 out of 20 false positive cases, presumed abuse should have been registered
- In 5 out of 20 false positive cases, there was reason for doubt, but she would have registered presumed abuse
- In 2 out of 20 false positive cases, there was no presumed abuse: one case of autism and one of severe speech delay
- In 5 out of 9 false negative cases, there was indeed presumed abuse, some cases being quite specific, but also a case of domestic violence
- In 4 out of 9 false negative cases, abuse had wrongly been presumed

Upon checking the actions that had been taken by the pediatricians, the liaison officer concluded that in many cases action had been taken accordingly but registration was flawed. This indicates that the problem is mainly with the registration of presumed abuse and with the ambiguity of the definition of abuse among the pediatricians. The review by the liaison officer proved for the JGZ that the model performs well enough to be useful in day to day operations.

5.15. Implementation

One of our goals in this research was to have JGZ's pediatricians work with our model to aid their day to day practice. We started with the simplest form of implementation by periodically running the model on an exported batch of children's files, and investigated the possibility of real-time decision support. The research director at GGD Amsterdam had this to say: *"This research shows that machine learning techniques perform well in predicting suspected child abuse. In practice we can benefit from using this model through implementation in a decision support tool, supporting our pediatricians in their judgment of a situation"*.

Together with the JGZ and the developer of their client management tool, it was decided to make the functionality of the prediction model available through an API (Application Programming Interface). The API needed to be called from within their existing software upon closing a child's file. A prediction is then made based on the entire file, including anything that had just been appended to it. The user is warned via a pop-up whenever a child's file is marked by the model as a case of possible abuse. This should urge the professional to take action and correctly register the case, if needed.

An important part of the implementation is the option for a professional to provide the model with feedback. This feedback is then used in the next learning cycle of the model and thus improves the model's performance in the long run.

The model ideally had to be implemented as a decision support system into the existing software used during the child consults. There are many different ways to have the model interact with the data and the pediatrician. One form of implementation was that directly after writing a new SOC, the data could be sent to an API (please see Appendix A) and the classification could then be presented to the user in the form of a warning whenever abuse is presumed. The subsequent action by the pediatrician could be recorded as feedback on the classification by the model. Using the feedback on these classifications, a training set contain-

[7] https://github.com/marcotcr/lime

ing less noise would be constructed on-the-fly, possibly improving the model's performance even further.

5.16. API development

We developed an API around the model following the REST (Representational State Transfer) architecture style. This way, the functionality of the model can be easily incorporated in any data management system that a client uses, while not requiring a direct connection to the client's database. For this we use the Flask[8] framework.

One endpoint ingests new data for scoring, that is all SOCs from the file and all structured variables used by the structured classifier in the ensemble. The API saves the data to a database which also contains training and test data. Next, the model makes and returns a prediction for the child together with the ID of the newly created database record. This ID is later used by the second endpoint to provide feedback on the prediction. Every night, the model is retrained using both the training and the feedback data. Fig. A.7 provides a schematic overview and illustrates these processes of predicting, getting feedback and re-training the model.

The API runs in a secure environment at the developer of the client management tool. Data is transferred and saved in anonymous form as much as possible, due to the sensitive nature of the data. On an average day, the medical professionals spread over 27 JGZ locations in this region of The Netherlands consult around 650 children between 0–4 (2014), amounting to around 650 new predictions per day.

6. Discussion

First, the research shows that classification algorithms outperform anomaly detection algorithms for the purpose of detecting presumed child abuse. Intuitively, this might be the result of not using the positive samples in training, losing a lot of information on "what qualifies as abuse". Additionally, child's file can be an anomaly for many reasons other than presumed abuse, e.g. a physical disability or a behavioral dysfunction. Due to underregistration there exist many positive samples in the negative group used for training the anomaly detection. This might severely limit the ability of the algorithm to identify the positive group as an anomaly.

From the analysis, it is evident that adding meta-features to the textual features improves the model's performance. Next, the best performances are attained when using a boosted Decision Tree algorithm like Random Forest (RF), or when using Support Vector Machine (SVM), that outperforms Naive Bayes (NB) mainly on recall. In order for the algorithms to deliver competitive results, we preselected a limited number of features using a statistical test, with χ^2 and ANOVA being equally good tests. This was not needed for SVM and might have even made the model performance worse; as SVM can deal with a very large number of features. Furthermore, the features needed to be weighted in any case using a tf-idf variant. We showed that in our case, a more advanced weighting schemes like DeltaTFIDF and BM25 did not outperform the standard tf-idf scheme.

The Area-Under-Curve (AUC) scores for the top RF and SVM classifiers were similar but indicated that a tuned SVM algorithm performed the best for the prediction of abuse from the unstructured data. This is in line with the majority of the text mining literature, that also propose SVM as the best choice algorithm. The SVM algorithm using a linear kernel with $C = 1$ and 1000 tf-idf weighted features performed the best for predicting child abuse from the unstructured data. The computed AUC was 0.906 with an accuracy of 0.843 and recall of 0.825.

A classifier based solely on structured data did not outperform the SVM classifier based on unstructured data. Combining the structured and unstructured data did, however, outperform the SVM classifier based solely on unstructured data. The best performance was attained when combining the two classifiers for unstructured and structured data into an ensemble method with an AUC of 0.914, an accuracy of 0.822 and a recall of 0.870.

Bellazzi and Zupan (2008) state that the project is not finished with a good model, one should ensure that the decision support system is properly implemented. This was also the goal of this research: to support the medical staff with the implemented model. To create impact with such an implementation, it was essential to let the client at the JGZ decide on the most appropriate form of implementation. Together with the JGZ and the developer of their client management tool, it was decided to make the functionality of the prediction model available through an API (Application Programming Interface). Since this research was first conducted at GGD Amsterdam, it has been repeated at three more GGD's. As evidence of evaluation of the API, a liaison officer in the first of these GGD's to actually implement the model said: *"We do not only use this model for signaling suspected child abuse, but it also helps to coach the pediatricians and helps them reflect on their own actions. This way, ideally, the model renders itself useless over time."*

The plan is to start the same research in five more GGD's in the Netherlands later in 2017.

7. Conclusions

In this article, we proposed a decision support system for identifying child abuse based on structural and free-text data. A systematic review of machine learning methods showed that the free-text data can indeed be used to signal presumed abuse when using classification algorithms. Both structured and unstructured data contain meaningful patterns that we used to create Random Forest and Support Vector Machine models. We achieved the highest score on the AUC-metric, which we identified as the most appropriate evaluation metric, by using an ensemble classifier combining the structured and unstructured data.

Our contributions to research and practice are: (i) Unlike previous studies, our study is based on a large dataset that is complete both in terms of quantity (all the children of the Amsterdam region over a 4 year time period) and quality (detailed information about every child included), (ii) Uniquely, this study takes both structured and unstructured data into account in building a prediction model, (iii) Moreover, in general, our model performs better than previous models (on our particular data) and we provide insight into the inner workings of the model (please see Section 5.13); and, (iv) Owing to the model's good performance it has been deemed useful for day to day operations, and we describe its implementation in a decision support system API for identifying child abuse in this study (see Section 5.15). Such an implementation, in our opinion, is a contribution to theory and practice, as it describes both the method of construction (of the underlying algorithms) and the deployment of the API.

The performance of the decision support system was not only evaluated mathematically but also by comparing its classifications with those made by the liaison officer from JGZ, an expert on child abuse. The high degree of agreement between this expert and our ensemble classifier lead to a wide acceptance of the proposed decision support system among the end users from the Dutch youth health care agency (JGZ).

This research has shown that utilizing machine learning techniques for children's health-related decision support is both feasible and beneficial. There are many ways in which this research

[8] http://flask.pocoo.org

can be further extended. Future work can focus on (i) Including more data from other relevant agencies, e.g. schools.(ii) Weighing evidence according to its temporal distance from the present moment.(iii) Extending the models for other threats to children's health.(iv) Evaluating the long-term effects of the automated identification of child abuse; and, (v) improving the understandability of the learned model itself for end users.

Our findings have the potential to improve the correct registration of child abuse, which is an important step to its prevention and the reduction of its effects.

Appendix A. API implementation

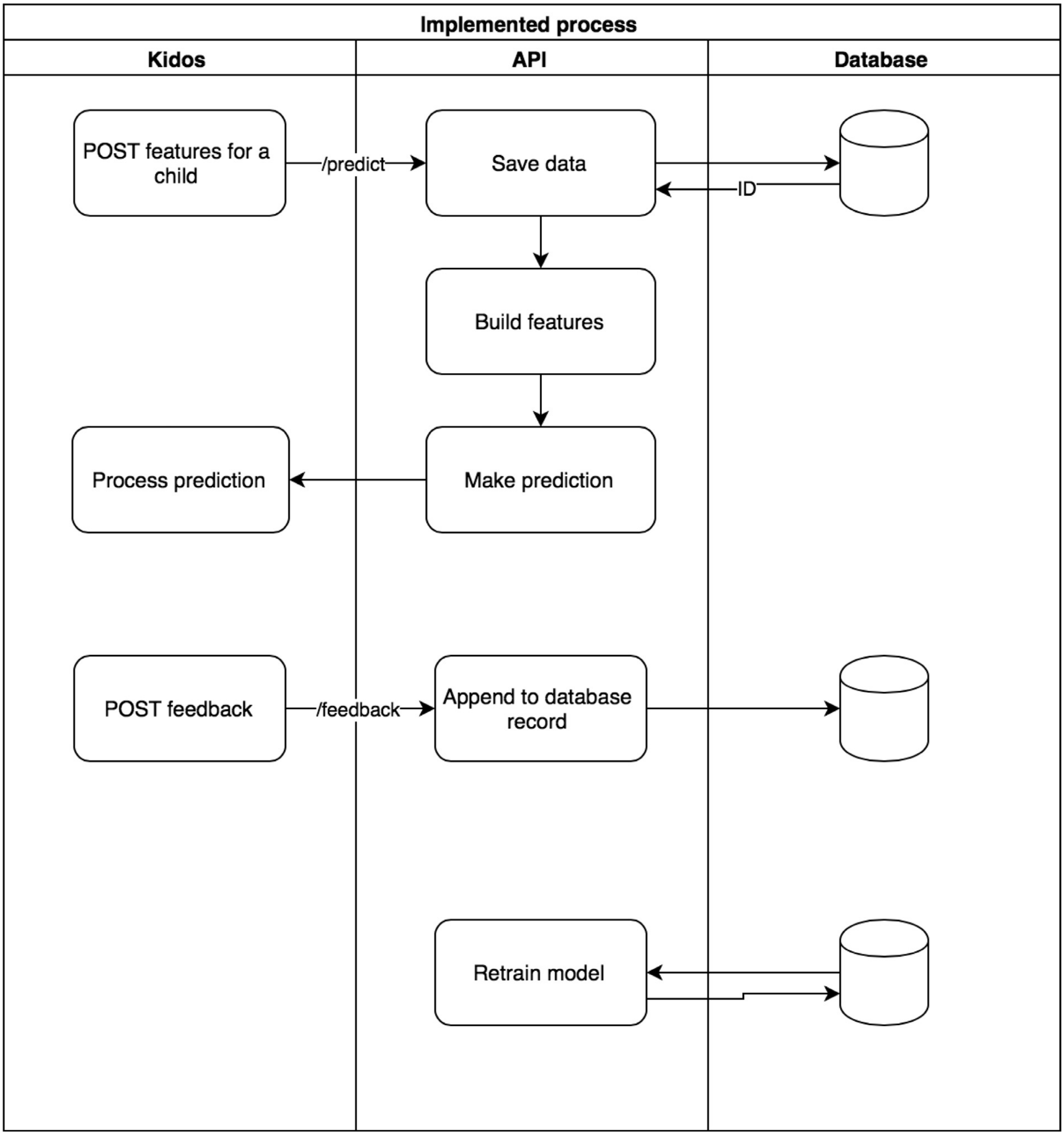


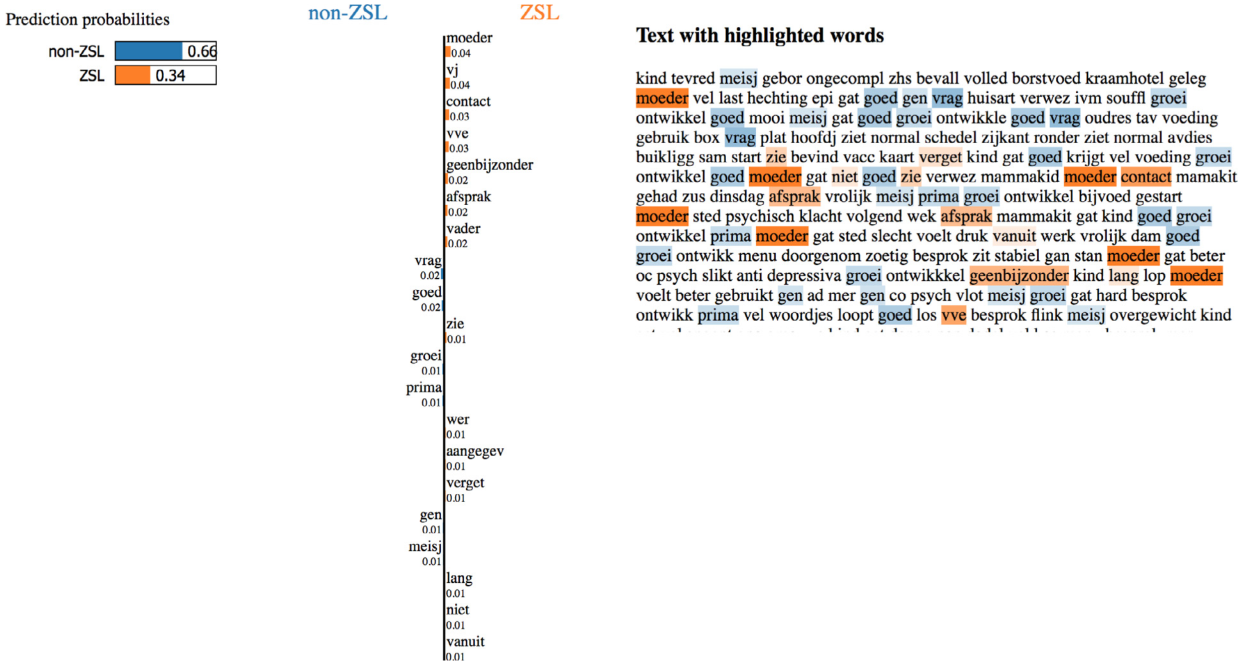
Fig. A7. Schematic view of the API implementation.

Appendix B. Initial layout

The data set from the GGD is split into several files, each having a specific subject. An overview of this data is included in Table B.12.

Table B12
Layout of the GGD data set.

File	Column	Type
Conclusions	Person number	integer
	Birth date	date
	JGZ location	text
	Action type	text
	Observation date	date
	Conclusion	text
Family relations	Person number	integer
	Child birth date	date
	Relation type	text
	Person number	integer
	Child relation birth date	date
BMI	Person number	integer
	Birth date	date
	Sex	text
	Action type	text
	Length	float
	Weight	float
	BMI date	date
	BMI age	float
	BMI	float
Worries ZSL	Person number	integer
	Birth date	date
	JGZ location	text
	Action type	text
	Observation type	text
	Value	text
Findings ZSL	Person number	integer
	Birth date	date
	Action type	text
	Finding date	date
	Finding type	text
	Finding	text
Actions ZSL	Person number	integer
	Birth date	date
	JGZ location	text
	Action type	text
	Observation type	text
	Action	text
Attention child	Person number	integer
	Birth date	date
	Attention	boolean



Text with highlighted words

kind tevred meisj gebor ongecompl zhs bevall volled borstvoed kraamhotel geleg moeder vel last hechting epi gat goed gen vrag huisart verwez ivm souffl groei ontwikkel goed mooi meisj gat goed groei ontwikkle goed vrag oudres tav voeding gebruik box vrag plat hoofdj ziet normal schedel zijkant ronder ziet normal avdies buikligg sam start zie bevind vacc kaart verget kind gat goed krijgt vel voeding groei ontwikkel goed moeder gat niet goed zie verwez mammakid moeder contact mamakit gehad zus dinsdag afsprak vrolijk meisj prima groei ontwikkel bijvoed gestart moeder sted psychisch klacht volgend wek afsprak mammakit gat kind goed groei ontwikkel prima moeder gat sted slecht voelt druk vanuit werk vrolijk dam goed groei ontwikk menu doorgenom zoetig besprok zit stabiel gan stan moeder gat beter oc psych slikt anti depressiva groei ontwikkkel geenbijzonder kind lang lop moeder voelt beter gebruikt gen ad mer gen co psych vlot meisj groei gat hard besprok ontwikk prima vel woordjes loopt goed los vve besprok flink meisj overgewicht kind

Fig. C8. Visual output of the LIME sensitivity analysis for end users.

Appendix C. LIME sensitivity analysis visual output

Fig. C.8 shows an example visual output for a single child's file. From left to right the prediction for this instance is first shown: in this case, the model assigns a risk of suspected abuse (ZSL) of 34% to this child. Next, the top 20 features influencing the predicted risk are listed, with the direction of influence. We can interpret this as the contribution of the features to the outcome. So, if this particular feature is removed, its contribution to the score, e.g. towards ZSL, should also be removed. The number of 20 top features is arbitrary. This can be used to highlight specific words or phrases in the text as having a positive or negative connotation. While interactions between the features are mostly ignored this way, the visualization sheds some light on the black box that is the text-based classifier. In this specific case we can see, in Dutch, that the mother has some psychological issues, but the child itself is doing well.

References

Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data* (pp. 163–222). Springer. http://link.springer.com/chapter/10.1007/978-1-4614-3223-4_6

Bellazzi, R., & Zupan, B. (2008). Predictive data mining in clinical medicine: Current issues and guidelines. *International Journal of Medical Informatics, 77*(2), 81–97. doi:10.1016/j.ijmedinf.2006.11.006.

Bressem, K., Ziegenhain, U., Doelitzsch, C., Hofer, A., Besier, T., Fegert, J. M., & Kuenster, A. K. (2016). A german e-learning-training in the context of early preventive intervention and child protection: Preliminary findings of a pre-post evaluation. *Child and Adolescent Psychiatry and Mental Health, 10*(1), 25.

Chapman, W. W., & Cohen, K. B. (2009). Current issues in biomedical text mining and natural language processing. *Journal of Biomedical Informatics, 42*(5), 757–759. doi:10.1016/j.jbi.2009.09.001.

Church, C. E., & Fairchild, A. J. (2017). In search of a silver bullet: Child welfare's embrace of predictive analytics. *Juvenile and Family Court Journal, 68*(1), 67–81. doi:10.1111/jfcj.12086.

Cios, K. J., & William Moore, G. (2002). Uniqueness of medical data mining. *Artificial Intelligence in Medicine, 26*(12), 1–24. doi:10.1016/S0933-3657(02)00049-0.

Cornet, R., Van Eldik, A., & De Keizer, N. (2012). Inventory of tools for Dutch clinical language processing. *Studies in Health Technology and Informatics, 180*, 245–249.

Demner-Fushman, D., Chapman, W. W., & McDonald, C. J. (2009). What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics, 42*(5), 760–772. doi:10.1016/j.jbi.2009.08.007.

Drucker, H., Wu, S., & Vapnik, V. N. (1999). Support vector machines for spam categorization. *Neural Networks, IEEE Transactions on, 10*(5), 1048–1054.

Fergusson, D. M., Grant, H., Horwood, L. J., & Ridder, E. M. (2006). Randomized trial of the early start program of home visitation: Parent and family outcomes. *Pediatrics, 117*(3), 781–786.

Friedman, C., Shagina, L., Lussier, Y., & Hripcsak, G. (2004). Automated encoding of clinical documents based on natural language processing. *Journal of the American Medical Informatics Association, 11*(5), 392–402. doi:10.1197/jamia.M1552.

Gillingham, P. (2015). Predictive risk modelling to prevent child maltreatment and other adverse outcomes for service users: Inside the black boxof machine learning. *British Journal of Social Work*, bcv031.

Goldman, G. M. (1990). The tacit dimension of clinical judgment.. *The Yale Journal of Biology and Medicine, 63*(1), 47.

Goryachev, S., Sordo, M., & Zeng, Q. T. (2006). A suite of natural language processing tools developed for the i2b2 project. *AMIA Annual Symposium Proceedings, 2006*, 931.

He, H., & Garcia, E. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*(9), 1263–1284. doi:10.1109/TKDE.2008.239.

Henry, S. G. (2006). Recognizing tacit knowledge in medical epistemology. *Theoretical medicine and bioethics, 27*(3), 187–213.

Ho, T. K. (1995). Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on: 1* (pp. 278–282). IEEE.

Horikawa, H., Suguimoto, S. P., Musumari, P. M., Techasrivichien, T., Ono-Kihara, M., & Kihara, M. (2016). Development of a prediction model for child maltreatment recurrence in japan: A historical cohort study using data from a child guidance center. *Child Abuse & Neglect, 59*, 55–65.

Hyun, S., Johnson, S. B., & Bakken, S. (2009). Exploring the ability of natural language processing to extract data from nursing narratives:. *CIN: Computers, Informatics, Nursing*, 215–223. doi:10.1097/NCN.0b013e3181a91b58.

Jones, E. D., & McCurdy, K. (1992). The links between types of maltreatment and demographic characteristics of children. *Child Abuse & Neglect, 16*(2), 201–215. doi:10.1016/0145-2134(92)90028-P.

Keddell, E. (2014). The ethics of predictive risk modelling in the aotearoa/new zealand child welfare context: Child abuse prevention or neoliberal tool? *Critical Social Policy*. 0261018314543224

Kononenko, I. (1993). Inductive and Bayesian learning in medical diagnosis. *Applied Artificial Intelligence an International Journal, 7*(4), 317–337.

Longfield, A. (2015). Protecting children from harm: A critical assessment of child sexual abuse in the family network in england and priorities for action. http://www.childrenscommissioner.gov.uk/sites/default/files/publications/Protecting%20children%20from%20harm%20-%20full%20report.pdf.

Malterud, K. (2001). The art and science of clinical knowledge: Evidence beyond measures and numbers. *The Lancet, 358*(9279), 397–400.

Manning, C. D., Raghavan, P., & Schtze, H. (2008). *Introduction to information retrieval*: 1. Cambridge university press Cambridge.

Martineau, J., & Finin, T. (2009). Delta tfidf: An improved feature space for sentiment analysis.. *ICWSM, 9*, 106.

Paltoglou, G., & Thelwall, M. (2010). A study of information retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 1386–1395). Association for Computational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others (2011). Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research, 12*, 2825–2830.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130–137.

Powell, C. (2003). Early indicators of child abuse and neglect: A multi-professional delphi study. *Child Abuse Review, 12*(1), 25–40. doi:10.1002/car.778.

Rao, A., Maiden, K., Carterette, B., & Ehrenthal, D. (2012). Predicting baby feeding method from unstructured electronic health record data. In *Proceedings of the ACM sixth international workshop on Data and text mining in biomedical informatics* (pp. 29–34). ACM.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 1135–1144). ACM.

Robertson, S. E., Walker, S., Beaulieu, M., & Willett, P. (1999). *Okapi at TREC-7: Automatic ad hoc, filtering, VLC and interactive track* (pp. 253–264). Nist Special Publication SP.

Schoech, D. (2010). Interoperability and the future of human services. *Journal of Technology in Human Services, 28*(1-2), 7–22.

Schoech, D., Jennings, H., Schkade, L. L., & Hooper-Russell, C. (1985). Expert systems: Artificial intelligence for professional decisions. *Computers in Human Services, 1*(1), 81–115.

Schrading, J. N. (2015). *Analyzing domestic abuse using natural language processing on social media data*. Rochester Institute of Technology Ph.D. thesis..

Sidebotham, P. (2003). Red skies, risk factors and early indicators. *Child Abuse Review, 12*(1), 41–45.

Spyns, P. (1996). Natural language processing. *Methods of information in medicine, 35*(4), 285–301.

Spyns, P., & De Moor, G. (1996). A Dutch medical language processor. *International Journal of Bio-Medical Computing, 41*(3), 181–205. doi:10.1016/0020-7101(96)01198-1.

Spyns, P., & De Moor, G. (1998). A Dutch medical language processor: Part II: Evaluation. *International Journal of Medical Informatics, 49*(3), 273–295. doi:10.1016/S1386-5056(98)00045-8.

Stacker, M. J. (2016). *Forensic interviewers' perspective of children who disclose abuse or neglect*. Capella University Ph.D. thesis..

U.S. Department of Health and Human Services, Administration for Children and Families, Administration on Children, Youth and Families, Children's Bureau (2016). Child maltreatment report 2014. Available from http://www.acf.hhs.gov/programs/cb/research-data-technology/statistics-research/child-maltreatment.

Vaithianathan, R., Maloney, T., Putnam-Hornstein, E., & Jiang, N. (2013). Children in the public benefit system at risk of maltreatment: Identification via predictive modeling. *American Journal of Preventive Medicine, 45*(3), 354–359.

Van Der Spoel, S., Van Keulen, M., & Amrit, C. (2012). Process prediction in noisy data sets: a case study in a dutch hospital. In *International symposium on data–driven process discovery and analysis* (pp. 60–83). Springer.

World Health Organization (2001). Prevention of child abuse and neglect: Making the links between human rights and public health. Available from https://www.crin.org/en/docs/resources/treaties/crc.28/WHO1.pdf.

Xu, H., Stetson, P. D., & Friedman, C. (2007). A study of abbreviations in clinical notes. *AMIA Annual Symposium Proceedings, 2007*, 821–825.

Yang, H., Spasic, I., Keane, J. A., & Nenadic, G. (2009). A text mining approach to the prediction of disease status from clinical discharge summaries. *Journal of the American Medical Informatics Association : JAMIA, 16*(4), 596–600. doi:10.1197/jamia.M3096.

Yoo, I., Alafaireet, P., Marinov, M., Pena-Hernandez, K., Gopidi, R., Chang, J.-F., & Hua, L. (2011). Data mining in healthcare and biomedicine: A survey of the literature. *Journal of Medical Systems, 36*(4), 2431–2448. doi:10.1007/s10916-011-9710-5.

Zeng, Q. T., Goryachev, S., Weiss, S., Sordo, M., Murphy, S. N., & Lazarus, R. (2006). Extracting principal diagnosis, co-morbidity and smoking status for asthma research: Evaluation of a natural language processing system. *BMC Medical Informatics and Decision Making, 6*(1), 30. doi:10.1186/1472-6947-6-30.

Zhou, X., Han, H., Chankai, I., Prestrud, A., & Brooks, A. (2006). Approaches to text mining for clinical medical records. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 235–239). ACM.

Zweigenbaum, P. M. (1995). Coding and information retrieval from natural language patient discharge summaries. *Health in the New Communication Age, IOS Press, Amsterdam*, 82–89.