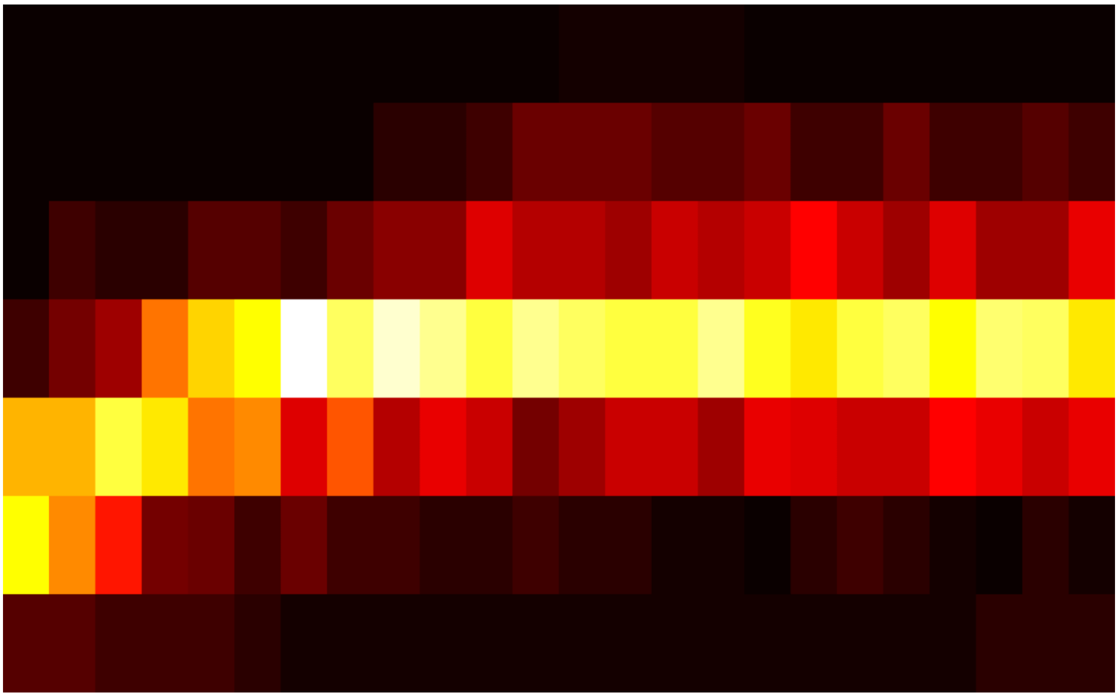

# Machine Learning for Technical Information Quality Assessment

*Master of Science Thesis in Computer Science - Algorithms, Languages, and Logic*

EMIL ANDERSSON
RICKARD ENGLUND

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
Göteborg, Sweden, March 2016

The Authors grant to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.
The Authors warrant that they are the authors to the Work, and warrant that the Work does not contain text, pictures or other material that violates copyright law.

The Authors shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Authors have signed a copyright agreement with a third party regarding the Work, the Authors warrant hereby that they have obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Machine Learning for Technical Information Quality Assessment

EMIL ANDERSSON
RICKARD ENGLUND

© EMIL ANDERSSON, March 2016
© RICKARD ENGLUND, March 2016

Examiner: DEVDATT DUBHASHI

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Cover: Visualization of word structural depth (vertical axis) and word bin index (horizontal axis), see page 23.

Department of Computer Science and Engineering
Göteborg, Sweden, March 2016

## Abstract

This thesis is about assessing the quality of technical texts such as user manuals and product specifications. This is done by consulting industry standards and guidelines, and implementing an automatic extractor for features describing the texts, based on these guidelines. These features are then put together into models, which are evaluated by using supervised machine learning algorithms on graded job application tests. Our conclusion is that it is probable that we can use this method and some of the features to classify the quality of technical texts. However, we think that it is hard to draw any confident conclusions using this small data set and suggest as future work to evaluate this on a larger data set.

# Acknowledgements

My first thanks goes to our supervisor Morgan Ericsson who has taken much time to meet, discuss and give feedback on our drafts. I would also like to thank Devdatt Dubhashi for taking his time examining our work and for his valuable mid-project feedback. Many thanks to Johan Thornadtson at Sigma Technology, who gave us the opportunity to sit in their nice office at Lindholmen and for giving us the data to work with. To my friend and co-author, Rickard, I would also like to express a thank you – it would have been more work and less fun without you and I probable couldn't have done it on my own. I am grateful for meeting so many lovely friends during these years at Chalmers, and I would especially like to thank Adrian. Last, but not least, I would like to thank Elin for your patience and support at home.

Emil Andersson, Chalmers, 2015

I would like to thank Morgan Ericsson, our supervisor, for his support and valuable input, Johan Thornadtson, at Sigma Technology, who has supplied us with the training data that we have used, Devdatt Dubhashi our examiner and last but not least Emil Andersson my co-author.

Rickard Englund, Chalmers, 2015

# Contents

# Chapter 1

# Introduction

An important part of the user experience for any technical product is the user manual and the technical specification. They should be the preferred method of support when a user needs help in what might be a stressful situation. Therefore, it is of utmost importance that the documentation is correct and easy to understand. For advanced systems, these texts often consist of several thousand pages of information and new versions are released several times each year. Their length and publication frequency makes it unreasonable to proof read them manually at a low cost.

Automatic methods exist for grading essays and other texts, but this project will focus on developing a method for classifying technical texts specifically. In essays a rich language is rewarded but in manuals a simple and easy to follow text is preferred. The reason simple language is preferred is that the target audience in many cases are non-native English speakers and it is quite probable they are under stress since they are in need of consulting the product documentation.

## 1.1 Background

*Sigma Technology* is a company that writes user manuals and they are interested in an automatic tool to estimate the quality of texts. Both in order for themselves to only deliver texts that are of a good standard, but also to be able to show that the quality of the manuals is improved after they have been edited by the company.

In order to find such a method there is a collaboration between *Sigma Technology*, *Gothenburg University*, and *Linnaeus University*, where they are searching for a way to automatically grade the quality of technical texts. In this process they proposed that machine learning might be a solution, which is what is evaluated in this thesis.

## 1.2 Problem

The purpose of this master thesis is to try to find an answer to the question: *How well can we grade the quality of technical texts using machine learning with graded job application tests from Sigma Technology as reference?*

The problem is divided into three parts. The first part consists of designing features we consider interesting, the second is combining these features into models, and the third part is about evaluating these models using supervised machine learning algorithms applied to job application tests from *Sigma Technology.*

## 1.3 Limitations

The texts processed in this work are of a specific type. We do not attempt to classify general texts. The texts considered are job application tests from Sigma Technology that are written in English. Even though the underlying goal is to classify manuals we will only use the job application tests in this first step.

Additionally, the work does not include unsupervised learning (since the premise includes graded training data), and the computational complexity of the algorithms is not a priority.

## 1.4 Related work

The areas relating to this thesis can be split up in two main areas; Text categorization and works on defining how to write technical English that is easy to understand.

### 1.4.1 Text categorization

Supervised text categorization is essentially what will be processed in this report. In this problem you have a set of categorized texts and you want to be able to find the category of other texts where you do not know the category. A subproblem to text categorization is automatic essay scoring. In automatic essay scoring you have a set of essays graded by one (or more) human grader(s). Then it is the computer's task to grade the essays. This area was proposed by Ellis Batten Page already in 1966 [1].

One example of work in automatic essay grading is the work performed by Larkey, L. S. [2]. Larkey used the $k$-Nearest Neighbor algorithm (see section 2.3.2) in order to classify essays. To quantify the texts they used a set of eleven features to quantify the texts. The features consisted of different length measurements of the texts, for example length in characters and average word length. Using this features they managed to achieve grading which correlated with a human grader on the same level as two human graders correlated with each other.

### 1.4.2 Technical English Writing

There have been several attempts to write guidelines on how to write technical texts in a way that is easy to use and understand.

*Simplified Technical English* [3] is a standard for how to write technical manuals by *AeroSpace and Defence Industries Association of Europe.* It is mainly intended for the aerospace industry but it is encouraged to be used in other areas as well.

*Ogden's Basic English* [4] is an attempt to create a simpler version of English Ogden claims to be able to express everything in English only using 850 different words. This language is supposed to be used international in business meetings and other such occasions where non-native English speakers are in need of communicating.

*Developing Quality Technical Information* [5] is collection of guidelines writing technical texts published as a handbook by IBM.

## 1.5 Outline

Following this introductory chapter is the Method chapter (Chapter 2) that describes the methods used to design features and models and to evaluate their performance. Then comes the Features & Models chapter (Chapter 3) that lists and explains the designed features and models. The evaluation results are presented and commented on in Chapter 4, but the tables containing the actual numbers are in Appendix C. The Discussion is in Chapter 5, and in Chapter 6 we try to draw conclusions from the results and point to possible future work.

# Chapter 2

# Method

This chapter presents the methods used in this work. It is divided into three parts related to the subproblems presented in Section 1.2. These parts are: designing the features (Section 2.1), combining the features into models (Section 2.2), and evaluating the classification performance of the models (Section 2.3).

## 2.1 Features

To try to find a measurement of the quality of a text, we look at a set of features describing the texts. This section discusses how we choose and design these features, while the actual features are presented in Section 3.1. The features are automatically extracted from the texts using a Java application we developed during this work. Some features are also further processed in MATLAB.

Our main source of inspiration on how a good technical text is composed is the book *Developing Quality Technical Information: A Handbook for Writers and Editors* [5], but we have also attended a crash course in Technical English [6] at Sigma Technology, read through the Simplified Technical English specification [3], and compared these sources with Sigma Technology's internal writing test grade description document used for grading the tests. In many aspects, these sources are all talking about the same concepts, but we will mostly be referring to the handbook in this report.

When deciding on features, we want them to reflect simple concepts (inspired by Occam's Razor) and they should scale to texts of different lengths to be more universal. We have also made the decision to only include features that are based on advice or rules in these sources. This decision was made to make sure we only have features for which we have rationale for.

Some of the features will not involve the complete text but rather be specific to certain tags in the markup. When tag filtering is used, it will be clear from the feature descriptions in Section 3.1.

## 2.2 Models

Before trying to classify the texts, we group the features into different sets. We call these sets of features for models. The reason we do this instead of just giving all the features to the algorithms in a single model is because the weights of the features in a model are considered to be equal in the algorithms used (see Section 2.3.2). Therefore, we need another way of defining the importance of each feature and we do this by selecting which features to include in the models. To aid us in this process of deciding which features to include in the different models, we have defined four groups of models we are using: the Single feature models, the Feature group models, the All features model, and also the Brute force model selection. This section describes why we choose these groups of models, while the actual models we are using are defined in Section 3.2. In the work, the models are defined in MATLAB.

### 2.2.1 Single feature models

The first group of models is the group of models consisting of one single feature each. The features will thus be evaluated on their own and this might give some indication of which features are useful.

### 2.2.2 Feature group models

The second group of models is the group of models consisting of related features. Testing these groups on their own might give some indication to if the areas that relate these features are interesting.

### 2.2.3 All features model

Since we only include features we consider interesting in this work, we are interested in a model that tests all these features together.

### 2.2.4 Brute force model selection

Since we have rather small amounts of data and not very many features, we are able to do some brute force searching for the best combination of features. We try all combinations of features and then find which of them has the best performance according to the method described in Section 2.3. Unfortunately, we did not have the time and computing power required to evaluate all possible feature combinations, which is why we have limited the Brute force model selection to models consisting of two and three features each. We have also limited the Brute force model selection to only be run with the CCR performance evaluation method (Correct Classification Ratio, see Section 2.3.3) and then calculate the AUC values (Area Under Curve in Receiver Operating Characteristics, see Section 2.3.3) for the best performing models (measured in CCR), since our computation power is limited and AUC takes significantly longer than CCR to calculate. We are aware that there might be higher AUC values further down in the list of best performing models

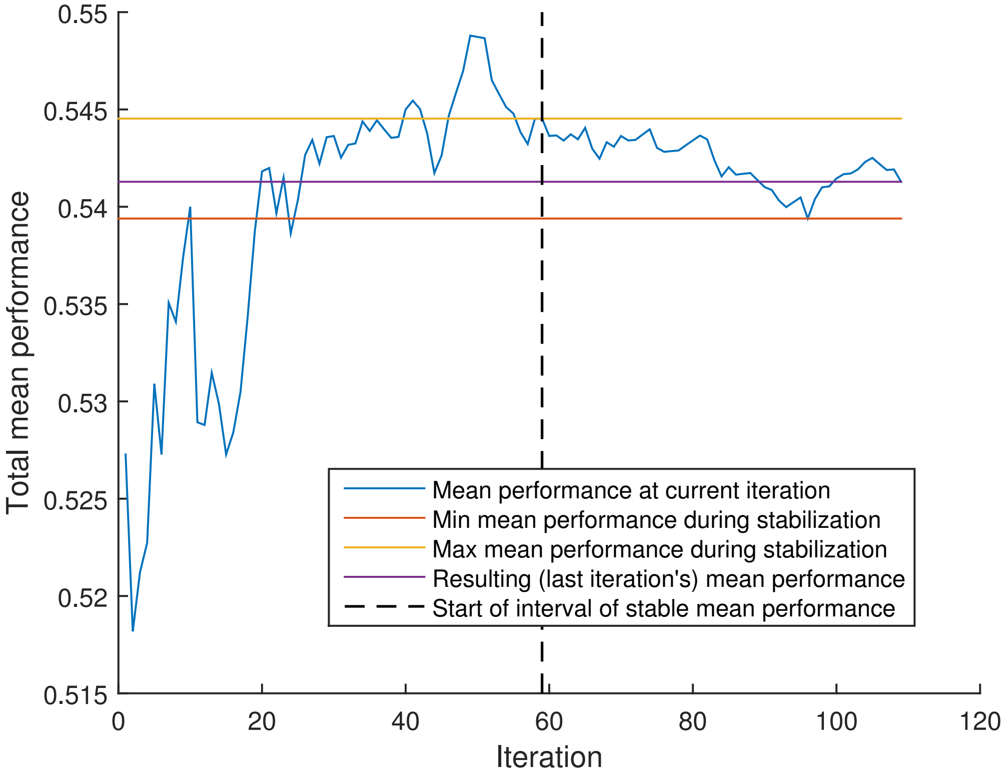

**Figure 2.1:** Performance stabilization example run.

measured in CCR, but we still consider it to be interesting to have a look at these five AUC values.

## 2.3 Evaluation

The evaluation of the models together with the classification algorithms (Section 2.3.2) is done in MATLAB using $k$-fold cross validation with the writing tests (Section 2.3.1) as training and validation data. The descriptive and the instructional writing tests (see Section 2.3.1 for an explanation of the two kinds of tests) are separated and run on their own, since they are quite different in their form. In the cross validation, the performance is measured using the measurements presented in Section 2.3.3. To get stable values, the cross validation is repeated (using new folds) until the mean of the performance values from all iterations settles near a horizontal asymptote. See Figure 2.1 for an example run where the total mean stabilizes at the end. For each model, algorithm and performance measurement combination, we store this combination's best performance together with any varied parameters for the algorithms used to achieve this performance.

We do not try to classify into all possible grades, but have instead separated the grades into the two classes *good* and *bad*. The reason for doing this is that we would have very few sample points for each class if we were to use all the grades as classes. We also consider binary classification as a simpler problem than multiclass classification. Since you need the grade 3 to pass the test, we chose to have the tests with grades above or equal to 3 in the good class and those with lower grades in the bad class. We also realized that some of the tests graded 2.5 were closer to 3 and some were closer to 2, which is why we made a separate set without these tests. These two sets are called the *complete set* and the *reduced set*.

For the $k$-fold cross validation, we use $k = 5$. The reason for choosing this value for $k$ is that it is a number that gives us rather much data to train on while still having some data points available to use for testing. For example, the AUC performance measurement requires at least one data point from each class to be computable. On the other hand, we want to train on as many data points as possible to get a reasonably well trained classifier, especially since our data set is quite small, but we do not want to overfit to the training data either. Choosing $k = 5$ seemed a reasonable compromise that for the complete sets gave 44 samples of training data and 11 samples of validation data.

Regarding the performance stabilization, we wait until the total mean performance value does not differ more than 0.005 for at least 50 iterations. These values were chosen after running the same evaluations multiple times and checking that the results were not varying much. The highest observed difference between these runs was within 0.01, which we deem to be stable enough.

### 2.3.1 Training Data

We received job application tests from *Sigma Technology* to use as our training data. We have also signed a non-disclosure agreement with the company, which is why we cannot reveal all details about the tests or present any concrete examples from them. We do not think this is a problem, and we present the interesting parts summarized below.

In total, we received 124 tests, written at various Sigma offices in the world but most of them have been written in Sweden. The tests are split up in two different types: *instructional* tests and *descriptive* tests, where each set contains 62 tests. The tests are different in structure, where the instructional tests are more structured while the descriptive tests are quite flat. The average length of each test is about 300 words.

The tests are handwritten and needed to be digitized to be usable in this project. We tried parsing the tests using OCR techniques but with very poor results, which is why we had to manually digitize them. Each text has been digitized by one of us, and we have together successfully entered 55 tests from each type. The reason that not all tests were converted is that the quality in the handwriting and the scanning among the writing tests varies greatly, some of the texts were close to impossible to read. We have done our best to parse what the authors intended to write, and when we were not sure what was written we gave the authors the benefit of the doubt.

The tests were digitized into an `XML` format inspired by `HTML`. The main reason for choosing an `XML` format was that the actual user manuals Sigma Technology are writing

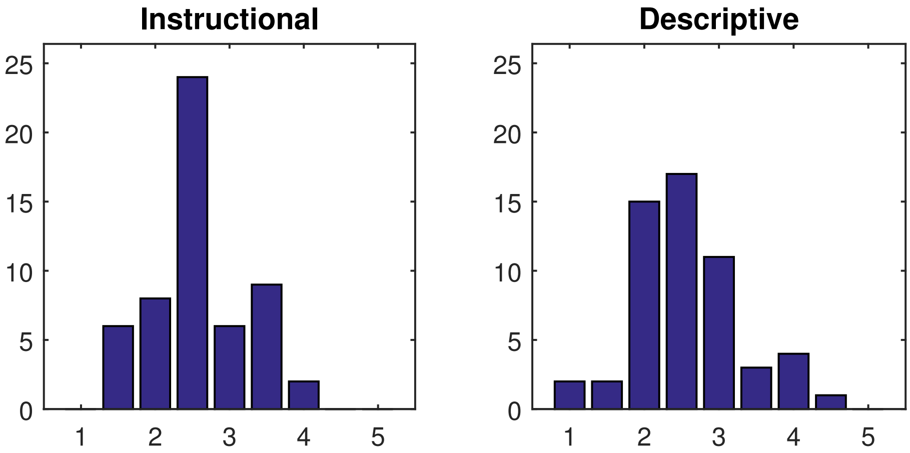

**Figure 2.2:** Grade distributions for the digitized tests.

are in an `XML` based format. It is also a common format that is easy to generate and parse. The format definition is available in Appendix A.

The grading scale is in the range from 1 to 5 where higher is better. There are also half grades when there is potential for a higher grade with some assistance. To pass the test, you need to get the grade 3 or better. Some tests also got grade 0, e.g. if they were completely unreadable or had too many flaws to be graded. None of the texts we managed to digitize had grade 0. Two persons have been responsible for grading the tests, but each individual test has only been graded by one person. The grade distribution for the digitized tests is shown in figure 2.2.

### 2.3.2 Algorithms

This section presents the different algorithms used in the evaluation process. The algorithms tested in this thesis is a subset of the algorithms from the paper *A re-examination of text categorization methods* [7] where they compared some different machine learning algorithms for text classification. Their result was that the algorithms were comparable except when the number of features was high. In that case the support vector machine algorithm outperformed the others.

#### $k$-Nearest Neighbors ($k$NN)

We use $k$NN [8] because it is a simple algorithm, and that it seems to work well even in complex situations with multiple clusters. Our implementation of $k$NN is based on the `knnsearch` [9] function from the *Statistics Toolbox* in *MATLAB*. If there is a tie, we choose which class to assign to the text uniformly randomly among the classes in the tie. The distance function used is simple euclidean distance. Since not all features are in the same range, we standardize the input so that each feature is centered at zero and scaled to have standard deviation one. This is done to make the weight of different features equal.

We vary $k$ from 1 to 39. The reason we chose 39 is that using the two classes bad and good with the complete training data sets, we got 18 good samples and 37 bad samples out of the 55 tests of the instructional type, and 19 good and 36 bad for the descriptive type. In a two-class $k$NN classifier, the result has the possibility to switch outcome until two times the size of the smallest class plus one of the nearest neighbors have been checked, because by looking at this many neighbors, it is impossible for half of the neighbors or more to be in the small class. This value is 37 for the instructional training data and 39 for the descriptive type, which is why we chose 39.

#### Support Vector Machines (SVM)

The second algorithm we use is SVM [10]. It is a commonly used algorithm and thus interesting for us to try as well. The implementation of SVM that we use is a standard implementation in *MATLAB*, `fitcsvm` [11] from the *Statistics Toolbox*. The kernels we use are the *linear* and the *radial basis function* kernels. The input is standardized in the same way as with $k$NN, which is also recommended as good practice by the documentation for the *MATLAB* function. For the other parameters, we are using the default options.

### 2.3.3 Performance measurements

In this section, the two different performance measurements used are presented.

#### Correct Classification Ratio (CCR)

The Correct Classification Ratio (CCR) measurement is the number of correctly classified texts divided by the number of tested texts. This method does not take into account if the size of the classes are different. For example if one class is only 10% of the total number of samples, then the CCR for the classifier always guessing on the largest class (the “largest class classifier”) would be 90%. This is correct in 9 cases out of 10, but it is not a very good classifier. Therefore you will need to know the ratio between the classes in order for this measurement to have a meaning.

#### Receiver Operating Characteristic, Area Under Curve (ROC, AUC)

The Area Under Curve (AUC) measurement is the area under the curve you get if you draw the false positive ratio (FPR) against the true positive ratio (TPR) while varying the classifier threshold, as described in [12, p. 183]. This curve is commonly called the Receiver Operating Characteristic (ROC). This performance measurement is not sensitive to imbalances in class sizes in the same way as CCR, and the AUC value for the “largest class classifier” described above would be 0.50 regardless of the class distribution. We use the *MATLAB* function `perfcurve` [13] for the AUC computations. For the positive class, we choose the bad class, since we think the target of the classifier is to flag for bad texts rather than identify good ones.

The classifier thresholds used are the ratio of the $k$ nearest neighbors that are in the positive class for $k$NN, and for SVM we use the `logit`[1] function to transform its scores to a threshold between 0 and 1.

As a parenthesis, we would like to mention that for the AUC measurement, it actually does not matter which class we choose as the positive class in our case. Assume we use one of our two classes as the positive class and have the values $TP_1$ (true positive), $FP_1$ (false positive), $FN_1$ (false negative) and $TN_1$ (true negative) for the classifier at one certain threshold. We then change to use the other class as the positive class and get the values $TP_2$, $FP_2$, $FN_2$ and $TN_2$. The machine learning algorithms we use do not depend on which class has been marked as positive, which is why its output should not be any different between these two runs. Assume that these two results are for the same case. We then have $TP_2 = TN_1$, $FP_2 = FN_1$, $FN_2 = FP_1$, and $TN_2 = TP_1$, since true positive becomes true negative, etc. The false positive rate is calculated by using the formula $FPR = FP/(FP + TN)$ and the true positive rate is calculated by the formula $TPR = TP/(TP + FN)$. For the second case, the FPR can be rewritten as $FPR_2 = FP_2/(FP_2+TN_2) = FN_1/(FN_1+TP_1) = (FN_1+TP_1-TP_1)/(FN_1+TP_1) = 1-TP_1/(FN_1+TP_1) = 1-TPR_1$. In the same manner, we also have $TPR_2 = 1-FPR_1$. We can thus plot these four axes in the same graph (see Figure 2.3 for an example) and we easily see that the area under the curve is the same no matter which of our two classes we choose as positive.

[1] $1/(1 + e^{-x})$, implemented in fitcsvm [11]

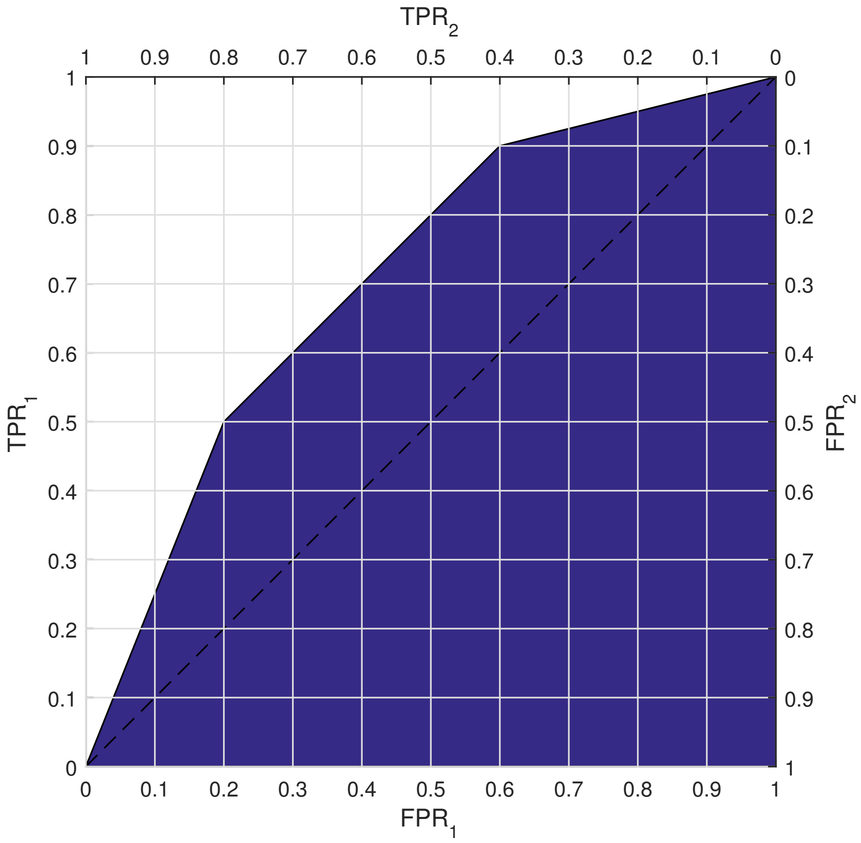

**Figure 2.3:** Illustration showing that the area under the curve (colored blue) is the same no matter which of our two classes we choose as positive. This is highlighted by the two sets of axes (corresponding to the different classes being chosen as the positive class) and their relation to each other.

# Chapter 3

# Features & Models

This Chapter presents all the features and models designed. The features are presented in Section 3.1 and the models are presented in Section 3.2. The methods used when designing the features and models are in Section 2.1 and Section 2.2 respectively.

## 3.1 Features

The features are grouped into different sets depending on what they are measuring. These groups correspond to subsections under this section. In each of these subsections, we present the features in that group together with explanations and motivations for having those features. In each subsection, all features in that group are also listed as reference.

### 3.1.1 Length features

Many of the guidelines are about keeping things short and simple, for example the advice: “Focus on the meaning” [5, pp. 105–109]. This advice is aimed at for example long sentences, imprecise words, unnecessary modifiers, and rambling paragraphs. This is why we have features based on counting the lengths of different parts of the text — the number of sentences per paragraph, words per sentence and characters per word. We also expand these to characters per sentence, characters per paragraph and words per paragraph to get some alternative ways of measuring the amount of content. One set of features is acquired by processing these six metrics by calculating their mean and variance for each text.

In addition to the mean and variance we have four features based on fixed numbers from the *LIX* readability index [14] and the *STE* (Simplified Technical English) specification [3]. The *LIX* readability index has defined that a word of more than 6 characters is a long word. *STE* states that when a sentence has more than 20 words it is too long for instructional texts, and that more than 25 words per sentence is too long for descriptive texts. In *STE*, it is also stated that a paragraph is too long if it has more than 6 sentences.

The parts of the texts that are measured in this set of features are only the parts that are in paragraphs (i.e. inside `<p>` tags), since we are looking at both sentences and paragraphs lengths.

To summarize, these are the 16 length features:

- Word Length In Characters - Mean & Variance
- Sentence Length In Characters - Mean & Variance
- Sentence Length In Words - Mean & Variance
- Paragraph Length In Characters - Mean & Variance
- Paragraph Length In Words - Mean & Variance
- Paragraph Length In Sentences - Mean & Variance
- LIX Long Words Ratio
- STE Long Paragraph Ratio
- STE Long Sentences Ratio Descriptive
- STE Long Sentences Ratio Instructional

### 3.1.2 Word etymology features

To find out what kind of language is used, we categorize the words into groups depending on their origin. In the handbook [5, pp. 127–128], they state that you often have a choice between two words with the same meaning but one is more direct. They also state that the more direct word is usually derived from Anglo-Saxon and the less direct word is usually derived from Latin.

Thus, we classify which words are derived from Latin or Anglo-Saxon words, using a Latin [15] and an Anglo-Saxon [16] list of words from *Wikipedia*. The extracted features are the relative usage of these two kinds of words, calculated by dividing the counts by the number of words checked.

Before trying to find the etymology of the words, we try to correct any probable spelling mistakes by using the spell checker in *LanguageTool* [17] (in the same way as in Section 3.1.5, where it is explained in more detail) and when a probable spelling error is found, we use its first suggestion for a replacement word instead of the probably incorrectly spelled word. The reason we do this is that the checks are dictionary based and any spelling mistakes will most certainly not be in these dictionaries. If the author meant to use a Latin word, this is what we should detect, regardless if it is correctly spelled or not. We are aware that the spell checker is not free from errors and that it is not always the first suggestion that is the word the author intended to write, but we consider this as a better solution than using the probably incorrectly spelled word.

When checking if a word is in the dictionary or not, we want to have the word in its base form since the dictionaries do not list all words in all forms. To do this, we use a process called lemmatization that tries to find the base form of a word. The lemmatization algorithm we use is based on an implementation in the *Java API for*

*WordNet Searching (JAWS)* [18] API for *WordNet* [19]. We also use the *Stanford Parser* (which is described in more detail in Section 3.1.4) to get the parts of speech of the words to faciliate the lemmatization process.

The parts of speech we can perform lemmatization on are verbs, nouns, adjectives and adverbs. If the word is not in any of these parts of speech, we just return the word as it is. If the parser finds that the word is already in base form (singular for nouns, base form for verbs, and positive for adjectives and adverbs), we also return the word as it is. Otherwise, we use the lemmatization function in *JAWS* to get a list of lemma candidates for this word with this part of speech. We then go through these candidates to see if they are present in the *WordNet* database with this particular part of speech and, if so, we add them to the result set of probable lemmas.

To summarize, these are the two word etymology features used:

- Anglo-Saxon Etymology Word Ratio
- Latin Etymology Word Ratio

### 3.1.3 Basic English features

To keep the text easy to understand it is reasonable to assume that the words used should be easy to understand. One part of the *Simplified Technical English* (STE) specification [3] lists allowed words. Additionally, the “Clarity” chapter in the handbook [5, pp. 103-146] is about making your texts clear and easy to understand, in many cases in terms of choosing the right words. The two features we have designed in this group are presented below. Since both are dictionary based, we correct any spelling mistakes in the same manner as presented in Section 3.1.2. For reference, the two features are:

- Ogden’s Basic English Word Ratio
- STE Approved Word Ratio

#### Ogden’s Basic English

The first Basic English feature is extracted by counting the number of basic words in a text. We implement this using the word list from *Ogden’s Basic English* [4], which is an attempt to create an universal language by using only a very small part of the English language. They claim that they can represent 90% of all English words by using their subset of only 850 words. The version we have downloaded contains all word forms of every word, so there was no need to use stemming for converting the analyzed words to base form. The resulting feature value is the ratio of basic words among the words checked.

#### Simplified Technical English feature

The *Simplified Technical English* (STE) specification [3] contains a dictionary that lists words that are approved and words that are not approved together with their part of speech. In the specification, it is also stated that “If a word is not in the STE dictionary,

| STE part of speech tag | Penn part of speech tag | Note(s) |
|---|---|---|
| v(erb) | VB, VBD, VBG, VBN, VBP, VBZ | - |
| n(oun) | NN, NNP, NNPS, NNS | - |
| pn (pronoun) | PRP, EX | - |
| art(icle) | DT | [1] |
| adj(ective) | JJ, JJR, JJS, PRP$, DT | [2] [3] |
| adv(erb) | RB, RBR, RBS | - |
| pre(position) | IN, TO | [4] |
| con(junction) | CC, IN | [5] |

**Table 3.1:** For each of the STE part of speech tags, this table shows which Penn part of speech tags maps to that STE part of speech tag. The Penn part of speech tags are presented in Appendix B. The notes are in the footer.

it is not approved (unless it is a Technical Name or a Technical Verb)". [3, p. 93] Since non-approved words can be both listed as not approved and not listed at all, we only look at if the words are in the list of approved words. We have not implemented any way of detecting Technical Names or Technical Verbs, and are simply ignoring this part of their rule. The dictionary also contains some phrases (e.g. "put on (v)", and "as to (pre)") and prefixes (e.g. "post-", and "pre-") that we have not implemented checks for. We consider this as a good feature even without these extra parts.

We use the *Stanford Parser* (which is explained in more detail in Section 3.1.4) to get the part of speech tags (listed in Appendix B) and then convert them to the smaller set of part of speech tags used by the Simplified Technical English specification in the way described in Table 3.1. We then check if the word together with its STE part of speech tag is in the list of approved words. Words with tags that are not in the parts of speech tag converstion table are not checked, e.g. the word three that has the CD (cardinal) tag. If a word has a Penn part of speech tag that can be in multiple STE part of speech tags (e.g. IN), all these are tested until a match is found. The resulting value measured is the ratio of the words checked that are in the approved list of words.

Regarding the forms of the words used in the dictionary and whether to use lemmatization or not (in the same way as described in Section 3.1.2), the specification [3, p. 93] states:

[1] In the Penn tags, the articles (a, an, ...) are listed as determiners.

[2] In the STE specification, the possesive pronouns are listed as adjectives.

[3] Some of the determiners (all, another, each, ...) in the Penn tags are listed as adjectives in the STE specification.

[4] In the Penn tags, prepositions are tagged together with subordinating conjunctions, which is why there is no direct mapping from stanford tags to STE parts of speech. However, we can search to see if the word is approved with this tag.

[5] It seems that all Penn conjunctions are conjunctions in STE as well. However, not all STE conjunctions are Penn conjunctions since some are in the prepositions group.

- “Nouns are shown only in their singular form, but plurals are permitted (unless a note tells you otherwise).”
- “Verbs are shown in the forms that are permitted (refer to Part 1, Section 3). Do not use verbs in other forms.”
- “Adjectives are shown in their basic form, with their comparative and superlative forms, if permitted, in parentheses.”
- “Approved adverbs are listed separately. Do not use an adverb if it is not listed as approved.”

This is why we use lemmatization on nouns but not on verbs, adjectives and adverbs. We have ignored any extra notes about the words only being allowed in that form, since these notes were very infrequent.

### 3.1.4 Verb forms features

The verb forms features are based on the output from the *Stanford Parser* [20], which is a natural language parser. A natural language parser is a piece of software that given a sentence outputs a phrase structure tree representing the grammatical structure of that sentence. Some examples of these trees can be seen in the description of the imperative usage feature further down in this section. The *Stanford Parser* also outputs grammatical relations (referred to as typed dependencies), such as subject and object relations, as explained by the *Stanford Dependencies* [21] project. For further reading about the output from the Stanford Parser, please look up the sources referred to or have a look at the summarization in Appendix B. We use version 3.5.0 of the parser package and the `LexicalizedParser` Java class with the `englishPCFG.ser.gz` model. There are some other parsers and models available in the *Stanford Parser* package, but we have not tested them.

We are aware that the parser is not always correct, but we consider it to be good enough. Additionally, we think that the parser should be better at parsing correctly written texts than incorrectly written texts, since it has been trained on real published texts. This is why we try to correct any spelling mistakes in the same manner as with the features presented in Section 3.1.2.

The five verb forms features are presented in the subsections below. They are:

- Verb Past Tense Verb Ratio
- Verb Present Tense Verb Ratio
- Imperative Sentence Ratio
- Active Voice Sentence Ratio
- Passive Voice Sentence Ratio

### Verb tense features

In this set of features, we look at the verbs and what tense they are in. The rationale for this is the advice "use the present tense" [5, p. 198]. We use the part of speech tags output from the parser and consider the tags `VBP` (verb, present tense, not 3rd person singular) and `VBZ` (verb, present tense, 3rd person singular) to be in the present tense and calculate the ratio of these among all words tagged as verbs (which are the tags starting with `VB`). We also calculate the ratio of the `VBD` (verb, past tense) tag among all the words tagged as verbs as a measurement of the past tense.

We do not measure the future tense because we have not found any easy way of doing that. The future tense is expressed using the base form (`VB`) of the verb, but there are other forms of verbs that are expressed using the base form as well (e.g. infinitive and imperative). It is sometimes expressed using modals (`MD`), as in the tagged sentence `I/PRP will/MD do/VB that/DT`. However, there are also other forms such as in the tagged sentence `I/PRP am/VBP going/VBG to/TO do/VB that/DT`. We have not investigated this any further.

### Imperative usage feature

The STE specification states that "In an instruction, write the verb in imperative ('command') form." [3, Rule 5.4]. This is why we find it interesting to measure the ratio of the sentences that are in imperative. This is done by analyzing the parse tree that is output from the parser for each sentence.

Most definitions we could find for the imperative form is that it is a sentence expressed as a command, but it was hard to find a clear description that we can implement in our feature extractor. Therefore, the definition we used for the imperative form is that it is a command expressed using the base form of the verb and usually with no subject in the sentence [22]. If the parse tree's root node has one child and that child has type `S` (simple declarative clause) and that `S` child has a child of type `VP` (verb phrase) that itself has a descendant node of type `VB` (verb, base form) and the `S` child does not have a `NP` (noun phrase) child, we classify the sentence as imperative. If it is not classified as imperative and the `S` child has children of type `S`, these children are tested recursively in the same manner and if any of them are classified as imperative, the whole sentence is.

The reason we do these checks is that if the `S` node does not have a `NP` child, then there is no subject. Additionally, the `S` node needs to have a `VP` child to have a verb and that verb should be in base form to be in the imperative mood. We do the recursive checks because some sentences are put together of smaller phrases by conjunctions, and these are output as separate sentence children in the parse trees. It is not enough to just look at the verb and see if it is in base form, since the base form is also used in, for example, the future tense.

Some different examples of imperative sentences with their parse trees are presented in Figures 3.1, 3.2 and 3.3. Figure 3.4 shows an example of a non-imperative sentence. There are also some sentence structures that have the subject explicit but we think

```
(ROOT
  (S
    (VP (VB Keep)
      (NP (PRP$ your) (NNS accessories))
      (PP (IN with)
        (NP
          (NP (PRP you))
          (PP (IN at)
            (NP (DT all) (NNS times))))))
    (. .)))
```

**Figure 3.1:** Parse tree of the imperative sentence “Keep your accessories with you at all times.”. The S, VP and VB nodes used to detect the imperative sentence are marked as bold. Note that the S node has no NP child.

```
(ROOT
  (S
    (S
      (VP (VB Unloose)
        (NP (DT the) (NN cord))))
    (, ,)
    (CC and)
    (S
      (NP (PRP they))
      (VP (MD will)
        (VP (VB wrap)
          (S
            (NP (PRP you))
            (VP (VB round))))))
    (. .)))
```

**Figure 3.2:** Parse tree of the imperative sentence “Unloose the cord, and they will wrap you round.”. The S, VP and VB nodes used to detect the imperative sentence are marked as bold. Note that the root S node is not marked as imperative and that the check has recursed to its children.

perhaps should be classified as imperative anyway. Some examples are “You, go there!” and “You should go there!”. Using our definition for the imperative, these sentences are not in this mood, though.

When developing this imperative detection approach, we tested it using the examples at grammar.about.com [23] (in some cases with slightly altered punctuation). It correctly handles 13 of the 20 examples and most of the fails are because the Stanford Parser incorrectly parses some of the sentences.

```
(ROOT
  (S
    (VP
      (VP (VB Go)
        (ADVP (RB ahead)))
      (, ,)
      (VP (VB make)
        (NP (PRP$ my) (NN day))))
    (. .)))
```

**Figure 3.3:** Parse tree of the imperative sentence “Go ahead, make my day.”. The S, VP and VB nodes used to detect the imperative sentence are marked as bold. Note that the VB node is not a child of the first VP, but it is a descendant.

```
(ROOT
  (S
    (NP (PRP I))
    (ADVP (RB accidentally))
    (VP (VBD ate)
      (NP (PRP$ my) (NN dog) (NN food)))
    (. .)))
```

**Figure 3.4:** Parse tree of the non-imperative sentence “I accidentally ate my dog food.”. Note that the S node has an NP child and that the S node’s VP child does not have a VB descendant. These two reasons (on their own) make sure it is not classified as imperative.

#### Passive and active voice features

The passive and active voice features measure the ratios of the sentences that are in active or passive voice. The reasoning behind this is the advice “Use the active voice” [5, pp. 196–167]. We do this by retrieving the collapsed dependencies [21] (summarized in Appendix B) from the parser. If we find a `nsubjpass` (passive nominal subject) dependency, we classify the sentence as passive. If no `nsubjpass` dependency is found and we find a `nsubj` (nominal subject) dependency, we classify it as active voice. This approach is tested against some active vs. passive sentence examples at yourdictionary.com [24] and it worked for all 32 examples where only one needed to be slightly changed.

### 3.1.5 Grammar and spelling features

The grammar and spelling features handle the grammatical style and correctness of the texts. They are presented in the two subsections below, and they are:

- Spelling Errors Per Word
- Other Errors Per Word

- Contraction Word Ratio
- Genitive Word Ratio

**Error checking features**

The rationale behind the error checking features is the advice: "Use correct grammar" [5, p. 191] and "Use correct and consistent spelling" [5, p. 197]. We use *LanguageTool* [17] to count spelling errors and other errors detected by the tool. The features are the number of errors divided by the number of words checked. The errors are split up in two categories: spelling errors and other errors. Other errors can for example be, but is not limited to:

- "Three successive sentences starts with the same word."
- "Use past participle here."
- "Possible agreement error. Did you mean frogs instead of frog?"
- "Use a in place of an."
- "A more concise phrase may lose no meaning and sound more powerful."
- "Did you forget a comma after a conjunctive/linking adverb?"
- "Sentence begins with small letter."
- "Don't put a space before the full stop."

The spell check is applied to all the words in the text but the other errors check is only applied to pieces of text where a full sentence is expected. This means that the other errors check is not applied to headings. We have adjusted the dictionary to allow certain words that were common and correctly spelled in the training data but was classified as possible spelling mistakes by *LanguageTool*.

**Contractions and genitives features**

The contractions and genitives features calculate the ratio of words that are contractions (e.g. it's, they're) or genitives (e.g. Emil's, tables'). We do this by counting the number of words with apostrophes in them. Then, we use the *Stanford Parser* [20] (explained in more detail in Section 3.1.4) to find how many `POS` tags (genitive markers) there are and set the contraction count to the difference between these two values. The genitive count is simply the number of `POS` tags used. These counts are then divided by the number of words checked to get the resulting features.

The rationale behind including the contraction feature is the advice "Do not omit words or use contractions to make your sentences shorter." [3, Rule 4.2]. When using genitives, we have seen in the notes made by the graders that the *of construction* is often preferred to the *'s* suffix.

### 3.1.6 Structural features

One of the three parts of the handbook is the “Easy to find” part [5, pp. 221-333], in which they discuss different techniques of structuring your texts. The features presented in the sections below are mostly inspired by this part of the handbook. This list summarizes them:

- Figures Per Word
- Figure Text Figure Ratio
- List Items Per Word
- List Lengths Mean
- List Lengths Variance
- Lists Per Word
- Word Depth At Bin Index
- Word Depth Ratios
- Depth Items At Depth Per Word In Whole
- Depth Items Per Word

#### Figure usage features

The reasoning behind using the figure usage features is the advice “Use visual elements for emphasis” [5, p. 287]. We have two features regarding figures. The first feature in this group is Figures Per Word, which is the number of figures in the text divided by the number of words in the text. This represents how frequently figures are used in the text.

The second feature is Figure Text Figure Ratio, which is the number of figure texts used divided by the number of figures in the document. In the case that there are no figures, we get a division by zero, which is solved by setting the ratio to 1 since we have the same number of figures and figure texts. The other alternative is to use 0 which would mean that no figure texts were used for any of the figures. The reasoning behind using 0 is that a low figure text ratio is probably bad and not using figures is also bad according to the advice. However, we do set the ratio to 1 since the responsibility for measuring the number of figures is assigned to the Figures Per Word feature.

#### List layout features

The inspiration behind the list layout features is the advice “Keep lists short” [5, p. 129]. To evaluate the list usage we have the following features:

- The number of list items per list, mean and variance (all depths in same set)
- The number of lists divided by the total word count in the document
- The number of list items divided by the total word count in the document

**Word structural depth features**

The word structural depth features try to capture the abstract notion of good structure and make it quantifiable. The main inspiration of these features is the guideline "Provide helpful entry points" [5, p. 274], but the rest of the "Easy to find" part [5, pp. 221-333] is also inspiring.

These features are calculated by assigning a structural depth to each word. The depth starts at zero and increases with sections and list items. Sections are not part of the data format (Appendix A), but is defined as the contents between a heading of a certain depth and the next heading of that depth or lower (or the end of the text). The reason that both sections and lists increase the depth is that some authors structure their content using headings while others use nested lists. We consider both these ways as equal when it comes to the depth of the document.

These word depths are then processed into a feature by first normalizing the different document lengths by putting the word depths for each word in one of a predefined number of bins, such that the first bin contains the depth of the first word, the last bin contains the depth of the last word and the words in between are sequentially put into the bins in a linear fashion. If there are more bins than words in a text, the values for the empty bins are linearly interpolated between the preceding and following bins that have associated words. We then use these bin indexes together with the mean of the word depth values in each bin as the structural depth for that bin. Each bin word depth is considered a separate variable in this feature. We tested some different bin counts (5, 10, 25, 50, 75, 100, 150, 200, 300 and 1000) before we decided on using 25 bins after observing that it performed somewhat better than the other bin counts when evaluated as single feature models.

We also have a simpler feature that is the ratio of the total number of words that are assigned to the different depths.

It could also be interesting to see how many depth items there are compared to how many words there are in the text, since this would measure how much structure there is, normalized by the document length. One such feature is made up by counting the number of depth items (sections and list items) there are at each depth and dividing it by the total word count in the text, and another by summing these values for all depths which gives the total number of depth items divided by the total word count.

## 3.2 Models

In this section we present the different models that we are evaluating. They are chosen in the way defined in the Method Chapter, Section 2.2.

The *Single feature models* correspond directly to the features presented in Section 3.1, with one feature per model. The *All features model* combines all of these 39 features into one model. The best performing models from the *Brute force model selection* are not presented here, but in Section 4.9.

The *Feature group models* (sometimes referred to as designed models) are grouped within the areas in which they are presented in Section 3.1. No single feature models are defined here, as they are already being tested.

The length features are combined into the following models: All length features, All means and variances, Simple[6] means and variances, Simple means, Simple variances, Expanded[7] means and variances, Expanded means, Expanded variances, STE and LIX fixed limits advice descriptive, STE and LIX fixed limits advice instructional, STE fixed limits advice descriptive, and STE fixed limits advice instructional.

There is only one Word etymology features model, and it contains both the Anglo Saxon Etymology Word Ratio feature and the Latin Etymology Word Ratio feature. Likewise, there is only one Basic English features model, which is the combination of the STE Approved Word Ratio feature and the Ogden's Basic English Word Ratio feature.

Among the verb form features, we have defined these models: All verb forms features, Verbs past and present tense, and Passive and active voice. The grammar and spelling features are grouped into the following models: All Grammar and spelling features, Contraction and genitive features, and Spelling and other error features. Lastly, the structural features are combined into these models: All structural features, Figure features, List features, and Word depth features.

[6] Word Length In Characters, Sentence Length In Words, and Paragraph Length In Sentences

[7] Sentence Length In Characters, Paragraph Length In Characters, and Paragraph Length In Words

# Chapter 4

# Evaluation Results

In this chapter, we comment on and highlight parts of the results after running each model, algorithm and performance measurement combination on the instructional and descriptive texts, as described in Section 2.3. The tables containing the actual results are available in Appendix C, which is outlined in the same way as this chapter.

The results are presented grouped into sections by the groups presented in Section 3.2, in Sections 4.1 to 4.9. In each of these sections, we note interesting numbers, and compare the different models, text sets and algorithms. The chapter then ends with Section 4.10, in which we point out general observations about the results presented.

A general problem is that in order for the CCR score to be an improvement over the "largest class classifier" that simply classifies all of the samples to be in the largest class, the CCR value will have to be greater than about 0.67 for the complete sets and greater than about 0.50 for the reduced sets. Using AUC, the value needs to be higher than 0.50 for both sets. For both evaluation methods, the value 1.00 represents a perfect classifier.

Many of the $k$NN results using CCR on the complete sets suffer from what we call the "$k > 30$, CCR $\approx 0.67$ problem", which occurs when the best result for those models is about 0.67 and is acheived when $k$ is high. What happens is that this classifier becomes the "largest class classifier", since the $k$ nearest neighbors will include all of the members of the largest class and will thus be correct in the cases texts from this class are tested. Since 0.67 is a moderately good performance value for CCR and these models are in fact quite poor, this result will be the best for these models. The results for the reduced sets do not show this problem as often, since they only need to have a CCR higher than 0.50 to be better than the "largest class classifier".

## 4.1 Single feature models

In the complete instructional texts set, using $k$NN, the classification performance is around 0.70 in AUC for the best performing single feature models but most CCR measurements suffer from the "$k > 30$, CCR $\approx 0.67$ problem". Using SVM, the performance is similar but with fewer good single feature models. The best performing single feature model using $k$NN is the Anglo Saxon Etymology Word Ratio model, but several single feature models are close. With SVM, the best performing single feature models are from the structural features group.

The reduced instructional texts set shows a significant improvement compared to the complete set. This is especially true for the Anglo Saxon Word Etymology Ratio single feature model, where the AUC value rises from 0.72 (with $k$NN) and 0.62 (with SVM) to above 0.90 for both algorithms, and the CCR values rise from about 0.67 to above 0.80 for both algorithms. A number of other single feature models also get higher AUC values for both $k$NN and SVM. We also get rid of the "$k > 30$, CCR $\approx 0.67$ problem". Some new single feature models appear in the top of the lists, but most good models in the complete set have about the same performance in the reduced set.

Since the evaluation performance of the Anglo Saxon Etymology Word Ratio single feature model improved so much in the reduced set, we got interested in looking at a plot of it, see Figure 4.1. This figure shows that removing the texts graded 2.5 to form the reduced set makes it easier to classify which of the good and the bad texts each text belongs to, since the texts with grade 2.5 seem to span the entire range while the texts below and above this grade seem to be more separated.

When trying to classify in the complete descriptive texts set, the performance is above 0.67 for the nine best single feature models using $k$NN, but only two of the models classified using SVM get this high. The best performing model is the Active Voice Sentence Ratio single feature model, using both algorithms. The Word Length In Characters Variance single feature model is among the best performing models using both algorithms as well.

Using the reduced descriptive texts set, the performance is about the same as with the complete descriptive texts set but, using SVM, more models get an AUC performance above or equal to 0.67 than with the complete set. The Active Voice Sentence Ratio model is still in the top using the reduced set.

Comparing the performance on trying to classify the descriptive and the instructional texts, they are about the same using the complete sets, but using the reduced sets is a significant improvement for the instructional texts but not for the descriptive.

## 4.2 Length features models

In the complete instructional texts set, the performance is about 0.70 with both AUC and CCR for the best performing models from the length features models group, using both $k$NN and SVM, but the AUC performance with SVM is about 0.3 to 0.5 higher than with $k$NN. Using the reduced instructional texts set, the AUC performance is still around 0.70 for the best models using $k$NN and about 0.75 using SVM. Looking at the best performing single feature models with features from this group, they are Paragraph Length In Characters Mean (AUC 0.69) using $k$NN on the complete set, Word Length In Characters Mean (AUC 0.65) using SVM on the complete set, Paragraph Length In Sentences Mean (AUC 0.74) using $k$NN on the reduced set and Paragraph Length In Sentences Variance (AUC 0.81) using SVM on the reduced set.

When trying to classify the complete descriptive texts set, using $k$NN, the perfor-

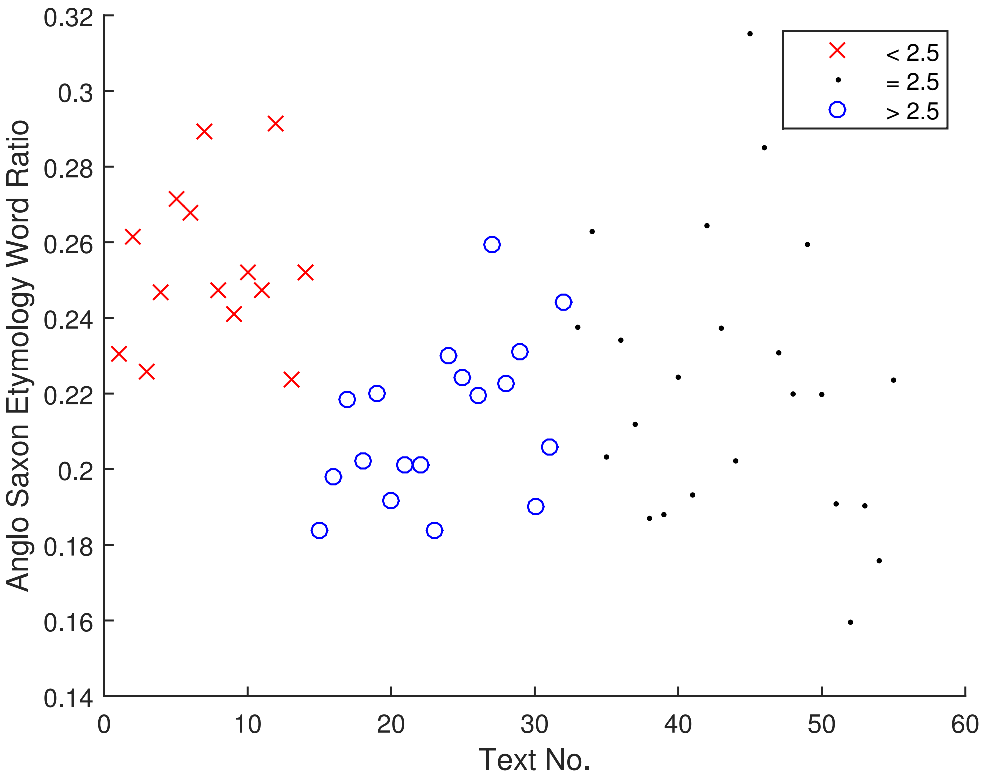

**Figure 4.1:** The Anglo Saxon Etymology Word Ratio for the instructional texts, divided into three groups to illustrate the change that happens when removing the texts graded 2.5 to form the reduced set. Note that the *Text No.* dimension (the x axis) is just for visualization purposes to be able to see each data point and that it is not present in any of the models.

mance is about 0.70 for the best performing features, with both AUC and CCR. The best performing model is the Simple variances model, which has an AUC value of 0.74. Using SVM, the AUC performance is below 0.70 for all models. Using the descriptive texts set, we get the opposite result to the instructional texts with a slightly, close to 0.05, higher result for AUC with $k$NN compared to SVM. The best performing single feature models for the descriptive texts are LIX Long Words Ratio (AUC 0.72) for the complete set using $k$NN, Word Length in Characters Variance (AUC 0.67) for the complete set using SVM, Paragraph Length in Characters Variance (AUC 0.72) with $k$NN on the reduced set and Sentence Length In Characters Mean (AUC 0.68) on the reduced set using SVM.

Comparing the classification performance of the instructional and the descriptive texts, we get about the same performance for the best classifiers. $k$NN seems to be better for the descriptive texts, but SVM seems to be better for the instructional texts. In all combinations of algorithms and text sets, some models perform around or slightly above 0.70 when measured using AUC.

## 4.3 Word etymology features model

In the complete instructional text set the combination model of the Anglo Saxon and the Latin word etymology features is always performing worse than the Anglo Saxon Etymology Word Ratio single feature model, but better than the Latin Etymology Word Ratio single feature model. As we can see in the evaluation of the single feature models, the AUC performance for the Anglo Saxon Etymology Word Ratio single feature model is 0.72 with $k$NN and 0.62 with SVM on the complete set. With the reduced instructional texts set, the AUC performance rises to above 0.90 using both algorithms and the CCR performance is above 0.80. The performance of the Latin Etymology Word Ratio single feature model is close to that of the "largest class classifier" for the complete instructional text set but somewhat better with the reduced set. With the combined model, the AUC performance rises from 0.58 with SVM and 0.69 with $k$NN for the complete set to 0.92 with SVM and 0.89 with $k$NN for the reduced set.

For the descriptive texts we get no results above 0.70 for any of the combinations of models, algorithms, and performance measurements, including the single feature models.

Comparing the complete instructional set and the complete descriptive set, the combined model and the related single feature models are performing at about the same level. When using the reduced sets, however, the classification performance on the instructional texts gets a significant improvement whereas the performance on the descriptive text remains at about the same level.

## 4.4 Basic English features model

Using the combined Basic English features model and the instructional text sets, we get an improvement from AUC performance at about 0.60 for $k$NN and about 0.50 for SVM to AUC performance at about 0.70 for both algorithms when using the reduced set compared to the complete set. For both algorithms, the CCR performance values are a bit lower with the reduced set than with the complete set. The reason for the improvement of the reduced over the complete instructional texts set seems to be that the STE Approved Word Ratio single feature model gets an improvement from using the reduced data set, but the performance of the Ogden's Basic English Word Ratio single feature model stays about the same.

When using the descriptive texts, the combined model performs close to the "largest class classifier" for both the reduced set and the complete set, and both of the algorithms. For both the descriptive text sets and both algorithms, the STE Approved Word Ratio single feature model outperforms the Ogden's Basic English Word Ratio single feature model in terms of AUC classification performance. However, it is not very good and has a maximum AUC performance value of 0.68.

## 4.5 Verb forms features models

For the complete instructional text set, we get AUC performance around 0.70 and CCR performance about 0.67 for all three models in the verb forms features models group. With $k$NN, we also note that $k$ is high for the combined models. The SVM algorithm has slightly poorer results than the $k$NN algorithm. The best performing single feature model from this group of features when evaluated on the instructional texts is the Verb Present Tense Verb Ratio single feature model in all four combinations of text sets and algorithms with an AUC performance above 0.70 for three of the four combinations.

With the complete descriptive text set we see about the same values but with slightly better results when using SVM and with less extreme values for $k$ when using $k$NN. The best performing single feature model from this group of features when evaluated on the descriptive texts is the Active Voice Sentence Ratio single feature model with an AUC performance at about 0.73 in all four combinations of text sets and algorithms. This is even the best single feature model among all the single feature models for all of these four combinations.

When using the reduced text sets instead of the complete text sets, we see that the All features model and the Passive and active voice model reaches an AUC performance at about 0.80 with the instructional texts, but we see no significant improvement in performance of the best performing model in the descriptive texts. Note that the best performing models are not always the same when comparing the different text sets, and that the difference we have looked at here is between the best performing model in each case.

## 4.6 Grammar and spelling features models

With the complete instructional texts set, the AUC performance is slightly above 0.60 for both algorithms for all models in the grammar and spelling features models group. Using the reduced set, though, we get a top AUC performance at 0.78 with SVM and 0.70 with $k$NN for the model combining all features in this group. Looking at the single feature models, the Spelling Errors Per Word single feature model was the best performing single feature model with features from this group in all four combinations of text sets and algorithms, and it is also improved when using the reduced text set compared to the complete text set with AUC performance values from 0.65 using $k$NN and 0.53 using SVM to about 0.75 for both algorithms.

For the descriptive texts, the AUC performance is around 0.50 for both algorithms and all models. This rises to about 0.60 in the reduced set with both algorithms and with a top result of 0.69 for the model combining all features in this group, using SVM. In three out of four of the text set and algorithm combinations, the Spelling Errors Per Word was the best performing single feature model with features from this group, but it is not very impressive with a maximum AUC performance at 0.64 for the reduced descriptive text set and using $k$NN as the classification algorithm.

Comparing the classification performance of the instructional and the descriptive texts, these models seem to be somewhat better at classifying the instructional texts, especially with the reduced set.

## 4.7 Structural features models

Using the complete instructional texts set, the evaluation results for the models in the structural features models group are all around 0.67 using the CCR performance measurement, but varies between 0.50 and 0.71 when using AUC. The best performing algorithm-model combination is SVM with the Word depth features model, which gets an AUC performance of 0.71 using the linear kernel. The worst performing model seems to be the List features model, which gets AUC performance at 0.57 using SVM and 0.50 using $k$NN.

With the reduced instructional texts set, the performance of the best performing model with each algorithm increases with about 0.05 compared to with the complete instructional texts set. Note that the best performing model is not always the same with the reduced set as in the complete set, and that the List features model gets an AUC value that is about 0.20 higher. The CCR values are not very impressive using the reduced set, but the best performing model gets a CCR value at 0.70. Looking at the single feature models with features from this group, the best performing single feature models are Figures Per Word (AUC 0.70) for the complete instructional set with $k$NN, Word Depth At Bin Index (AUC 0.71) for the complete instructional set with SVM, Lists Per Word (AUC 0.81) for the reduced instructional set with $k$NN, and Depth Items At Depth Per Word In Whole (AUC 0.79) for the reduced instructional set with SVM.

On the complete descriptive texts set, the CCR values are about 0.67 for both algorithms and the AUC values vary from 0.52 to 0.62 with $k$NN and are all at about 0.50 with SVM. All models seem to be performing quite poorly and no model seems to be significantly better than any other. Using the reduced set, there is no significant improvement, either.

Comparing the instructional and the descriptive texts, these models seem to perform better on the instructional texts, especially with the reduced set.

## 4.8 All features model

The all features model (which includes all of the 39 features) performs the best on the reduced instructional texts set using $k$NN. Using this configuration, the AUC performance is 0.75, which is about the same performance as many of the best manually designed models. On the descriptive texts, this model performs best using SVM on the complete set, where we get an AUC value of 0.71. Comparing the instructional and the descriptive texts, it seems to be somewhat better at classifying the instructional texts than the descriptive, especially using the reduced texts set.

It is interesting to note that the All features model achieves about the same performance as the best single feature model in three of the eight text sets and algorithm combinations: the complete instructional set using $k$NN, and the complete descriptive set using $k$NN and SVM.

## 4.9 Brute force model selection

In this section we comment on the top five brute force models for the different instructional and descriptive text sets and the two algorithms used. Note that there might be AUC values higher than the ones presented here, since we have only done the AUC measurements for the best performing models measured by CCR, and that when we mention the best brute force models, we mean the best brute force models measured in CCR.

For the instructional texts with the complete text set and with brute force models consisting of two features each, we get results with both AUC and CCR performance close to 0.80. The values for SVM are similar but they are using different features. Using $k$NN, all of the $k$ values are 1 except for one. When increasing the number of features per model to three, the CCR values are increased from slightly below 0.80 to slightly above 0.80. Using SVM, we even get some models with an AUC value as high as 0.87. Looking at the top five performing models for these four combinations of algorithms and features per model, we get a total of 20 high performing models. In these models, features from the Length features group are present in 16 out of the 20 models. The other groups of features that are also represented in these high performing models are Verb forms features, Structural features, Word etymology features, and Grammar and spelling features. The Basic English feature group is not represented at all in these 20 models. The kernel in the SVM algorithm is linear in only 3 out of these 20 models.

When we change the text set to the reduced instructional text set, we get even more improved test results with CCR values around 0.85 for the best models consisting of two features each and around 0.90 for the best models consisting of three features each. Some models even have AUC values above or equal to 0.95. With the reduced text set, we can see that the Anglo Saxon Word Etymology feature is part of 17 out of the 20 high performing models mentioned in the previous paragraph, which reflects the increase that this feature gets in the evaluation of the single feature models as well. We can also note that none of the Structural or the Basic English features are in any of these 20 high performing models with the reduced instructional text set.

Using the complete set of descriptive texts, we get CCR values around 0.75 for the best models using two features, which increases to 0.80 when using three features. The feature Active Voice Sentence Ratio is in 18 out of the 20 high performing models (as mentioned previousle in this section), and the feature Paragraph Length In Characters Variance is in all of the ten high performing three feature models. All of the 20 high performing brute force models found when running the evaluation on the complete descriptive texts set only use features from the three groups: Length features, Verb form features, and Structural features. The linear kernel is used for SVM in only 1 out of the 20 models.

When using the reduced descriptive text set, we get CCR values similar to the ones

| Feature | Occurrences |
|---|---|
| Anglo Saxon Etymology Word Ratio | 20 |
| Spelling Errors Per Word | 18 |
| Word Length In Characters Mean | 13 |
| Sentence Length In Words Mean | 6 |
| Verb Present Tense Verb Ratio | 5 |
| Sentence Length In Characters Mean | 5 |

**Table 4.1:** This table shows all the features which occurred at least five times in the top five best performing brute force models for the instructional texts, for all combinations of two and three features, for both the complete set and the reduced set, and for both algorithms.

| Feature | Occurrences |
|---|---|
| Active Voice Sentence Ratio | 29 |
| Paragraph Length In Characters Variance | 20 |
| List Items Per Word | 5 |
| Lists Per Word | 5 |

**Table 4.2:** This table shows all the features which occurred at least five times in the top five best performing brute force models for the descriptive texts, for all combinations of two and three features, for both the complete set and the reduced set, and for both algorithms.

for the complete descriptive text set: About 0.75 for the best models consisting of two features and close to 0.80 for the best models consisting of three features. The best (measured with CCR) model consisting of three features when using SVM has an AUC value of 0.87. Compared to the complete descriptive text set, we also get some features from the Basic English features group in some of the 20 high performing models, and there are still features from the Length features group in 18 out of the 20 models. The feature Active Voice Sentence Ratio is also in 7 out of the ten high performining models consisting of three features.

In Table 4.1 and Table 4.2 we present, for the instructional and the descriptive texts respectively, the most common features that occurred at least five times in the top five best performing brute force models for all combinations of two and three features, for both the complete and the reduced set, and for both algorithms. Both these sets of results contain 40 results each.

Comparing the descriptive and the instructional texts, we see that the results from the instructional texts are improved when using the reduced test set, but for the descriptive texts we can not see such an improvement. We can also see that different features seem to be more interesting for the different texts. Additionally, there are only 3 out of 80 high performing models that contain features from only one group of features, indicating that combining features from different groups probably is a good idea.

## 4.10 General observations

In this section, we try to highlight differences and similarities between the different groups of models presented in this chapter.

Looking at the designed models (which excludes the single feature models and the models from the brute force model selection), the reduced instructional texts set had a better classification performance than the complete instructional texts set, with 0.05-0.20 higher AUC performance values, for all groups of models except the Length features models and the Structural features models. In these two groups, the performance was about equal for both text sets. For the descriptive texts you could only see this kind of improvement when using the reduced set instead of the complete set for the Grammar and spelling features models group.

The best performing designed model (which excludes single feature models and the models from the brute force model selection) was the Word etymology features model with an AUC value around 0.90 and CCR performance close to 0.80 on the reduced instructional texts set. The All verb form features model was the second best performing model with an AUC value of 0.80 for the same text set. Most of the other groups of models had AUC values close to 0.70-0.75. Some of the groups of models had very low performance on descriptive texts, for example the Structural features models with AUC values close to 0.50 and CCR values around 0.67 or 0.50 depending on the data set.

Generally, most of the results do not show very impressive results, with CCR values close to the “largest class classifier”. We can however see that some of the models get better results, and that some of the features in these models keep reappearing in the best performing brute force models. We can also see that different features seem to be better at classifying the two different text types.

One interesting thing to note is that the structural features appear in the brute force models for the descriptive texts, which we considered to be flat documents without structure. However, there are actually ten descriptive texts that are using lists in the training data and these texts are probably the reason that the structural features appear also in the descriptive texts.

# Chapter 5

# Discussion

In this chapter we lift discussable parts of the work and try to identify threats to the validity in the methods used. The chapter is divided into two sections. In Section 5.1 we discuss the data used, and in Section 5.2 we discuss the features and the models designed.

## 5.1 Training Data

What we consider as the main issue with this work is that we have had access to a quite small data set, limited both by the number of texts we have had access to and by how many of them we have been able to digitize. We believe that this small data set makes it hard to get stable results and to try to draw any conclusions with confidence.

Another thing to consider when evaluating this project is the digitization process. The texts have been manually digitized by us, and even though we have tried to be as careful as possible when entering the texts, it is probable that we have made some mistakes, such as typographical errors. We have, however, gone through the spelling errors reported by the spell checker used and checked in the original texts if these errors are actually present and corrected those that were not. It is more probable that we have entered some words correctly spelled when they should have had spelling errors, since it is easy to happen to read the word intended even though it did contain spelling errors. Regarding the format used, we have tried to define a clear format to know how to parse the texts but, for example, the question of when a text is a paragraph or not does not have a completely obvious answer. In the end we decided that when a text can be seen as a distinct section it is a paragraph. As a result of this we got a large amount of short paragraphs in many of the instructional texts, which are often made up of lists with one sentence per list item. Lastly, each text has been entered by only one of us. It would have been beneficial from a correctness perspective to have both of us enter each text separately and then compare them to find the differences and hopefully spot mistakes, but that would have made this part of the work too time consuming. However, we do agree on the format used and we believe that we are interpreting it in the same manner. To sum this part of the discussion up, we are aware of potential problems in the digitization process but do not consider them as major concerns.

We also have to consider the grading process of the tests used in our training. Each test has been graded by only one grader. The different tests have been graded by two graders, and the graders would probably not do the exact same grading if both of them were to grade the complete set of tests separately. However, we have read the internal document outlining the grading process and we consider it as clear, and think that the two graders would probably have done roughly the same grading. We consider it as very improbable that one grader would assign the grade 4 to a test and that the other grader would assign it the grade 2, but more probable that one grader would assign a test the grade 2.5 and that the other grader would give it grade 2 or 3.

Regarding the grading, we can also discuss how we have converted the grades into the classes used for classification. One could argue that the tests with grade 2.5 perhaps should have been accepted as good tests. The grade 3 is the lowest grade needed to pass the test, but the grade 2.5 indicates that the writer has potential to reach grade 3 with some initial support. Our tests show that it seems to be easier to classify the instructional texts when removing the tests graded 2.5, but is possible that this is a coincidence. The reason we chose to try with this reduced set is that some of the tests graded 2.5 are probably closer to the grade 2, while some are probably closer to the grade 3. Thus, it seems reasonable to avoid this grade. It is possible that you could have achieved the same effect by removing random tests instead of the tests graded 2.5.

One could argue that the instructional texts are perhaps harder to grade and that when the grader is in doubt of whether to pass or fail a test they could perhaps tend to put 2.5 as the grade. This reasoning could lead to that the 2.5 graded texts are very noisy and that the decision to use the reduced set is a good idea. This could also be reflected in the grade distribution (Figure 2.2) where we can see that 2.5 is more than twice as common as any other grade, which is not true for the descriptive texts. However, the number of samples for both these text types is quite small with 55 samples each, and it is not very improbable that the underlying distribution is a normal distribution with a mean at 2.5. This would make sense, since most people taking the test would not be good enough candidates, a few would be very bad, and a few would be really good.

It could also be interesting to have more than two classes or perhaps a real valued output from the classifier instead of distinct classes. We decided early that we would use two classes in this work for simplicity and because we wanted a more feasible amount of data per class, but we think such different classifications should be considered as interesting future work.

## 5.2 Feature and model design

Regarding our features, we have made the limitation to only include features for which we can find advice to justify the feature. It is possible that we could have come up with other features which would turn out to work even better, but we wanted to have a reasonable source for why to include the features.

The LIX readability measurement index is constructed with Swedish in mind, and it is possible that the limit six characters or more for long words is more adapted to that language than to English. It could be interesting to use another value as a long word limit, but we are not certain what that value would be.

Regarding the word etymology features, we have used compilations of words originating from Anglo Saxon and Latin which were published on Wikipedia. It is probable that there are better sources available, such as lexicons with word etymologies included, but we consider these lists as good enough for testing the concept.

It is debatable whether the decision to use a spell checker to try to correct words that are probably misspelled before doing the dictionary checks and the sentence parsing was a good decision or not. We just use the first suggestion from the spell checker and have not analyzed how it weighs its suggestions. After checking some of the texts, we are also aware that it does not always replace the misspelled words with the word that was actually meant. It would be interesting to try analyze the sentence using the parser with all different suggestions from the spell checker to try to see which word would fit best. We have not checked if the Stanford Parser has the ability to return how probable a sentence would be, but if it has such an ability it would be an interesting future work. Another interesting spell correcting approach that would perhaps work better is a Bayesian spell corrector in the same manner as proposed by Peter Norvig [25], since it would be based on probabilities for its ranking of alternative words. Nonetheless, we think the decision to replace probably misspelled words with the spell checker's first suggestion was a good one and that it does more good than harm.

When combining the features into models and then running the machine learning algorithms on them, we have used algorithms that have equal weighting of all features in the model. Therefore, we have designed different models where each feature is either in the model or not, corresponding to the weights one or zero. We have also used brute force model selection to try to find which features perform the best together, but the features are still just included in the model or not. It would be interesting to try an approach where we can have different weights for the different features and find the best weights as well, but we consider our current approach to be good enough for now.

Regarding the brute force model selection, it could also be interesting to use some heuristic method to try to find the best performing feature combination faster. Early on, we tried the Add-One-In Analysis used in [26], but we could not get it to work properly and we could clearly find better performing models using brute force. This could be because most of the features had about equal performance when run as single feature models, or it could also be that it is not a very good heuristic. In the absence of a good heuristic, we believe that our method is good, but we have only had the computing power to test brute force models consisting of 2 or 3 of our 39 features.

# Chapter 6

# Conclusion & Future Work

This chapter contains our conclusion and tries to highlight possibly interesting future work. The conclusion is presented in Section 6.1 and the future work is presented in Section 6.2. Many of the points made here are also discussed in Chapter 5.

## 6.1 Conclusion

As we mentioned in Section 5.1, it is hard to try to draw any confident conclusion with this small data set. However, it seems like some of the features and models show a potential for being good classifiers. For example the Anglo Saxon Word Etymology feature performed well with the instructional texts, and some of the brute force models also achieve good performance. The performance of the models is also different depending on the text type, which is to be expected since the different texts are not very similar. Therefore, our conclusion is that it is probable that we can use this method and some of these features to classify the quality of technical texts, but that we need to train and validate on larger data sets.

## 6.2 Future work

Since we identified the small data set to be the largest issue with this work, our first suggestion for future work is to try this method and these features on a larger data set. With a larger data set, it would also be interesting to have completely separated data for testing, that is not even used in the $k$-fold cross validation. It would also be interesting to run the trained algorithms on actual manuals and see if the grading seems reasonable.

To try to improve the results, it is also possible to tweak the algorithms, where we have only used the default settings provided by *MATLAB*. It would also be interesting to try other machine learning algorithms and other ways of turning the grades into classes, as discussed in Chapter 5. We suggest these extensions as interesting future work.

It is always possible to try more features. You could, for example, try features that are based on the actual words used. We opted not to use these content based features, like counting the number of times a specific word is used, because these features would be very specific to exactly this training text (the tests within a text type are all about the same topic) and would probably not scale well to other texts about some other topic. However, this kind of features would probably give nice results for these specific texts.

If you have access to more computation power, or can come up with a good heuristic, it would be interesting to see how the brute force model selection would perform on models consisting of more features and also how large the models can be in terms of the number of features in them before the results would start to decline towards the model containing all features.

# References

[1] E. B. Page, The imminence of... grading essays by computer, Phi Delta Kappan (1966) 238–243.

[2] L. S. Larkey, Automatic essay grading using text categorization techniques, in: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 1998, pp. 90–95.

[3] ASD, Simplified technical english, asd-ste100 (2013).

[4] Ogden's basic english, [Online; accessed 9-December-2014].
URL http://ogden.basic-english.org/basiceng.html

[5] G. Hargis, Developing Quality Technical Information: A Handbook for Writers and Editors, IBM Press Series, Prentice Hall Professional Technical Reference, 2004.

[6] Sigma, Writing technical english, fundementals - crash course.

[7] Y. Yang, X. Liu, A re-examination of text categorization methods, in: Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '99, ACM, New York, NY, USA, 1999, pp. 42–49.

[8] T. Cover, P. Hart, Nearest neighbor pattern classification, Information Theory, IEEE Transactions on 13 (1) (1967) 21–27.

[9] The MathWorks, Inc., Find k-nearest neighbors using data - MATLAB knnsearch - MathWorks Nordic, [Online; accessed 5-March-2015].
URL http://se.mathworks.com/help/stats/knnsearch.html

[10] B. E. Boser, I. M. Guyon, V. N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92, ACM, New York, NY, USA, 1992, pp. 144–152.

[11] The MathWorks, Inc., Train binary support vector machine classifier - MATLAB fitcsvm - MathWorks Nordic, [Online; accessed 3-March-2015].
URL http://se.mathworks.com/help/stats/fitcsvm.html

[12] K. Murphy, Machine Learning: A Probabilistic Perspective, Adaptive computation and machine learning series, MIT Press, 2012.

[13] The MathWorks, Inc., Receiver operating characteristic (ROC) curve or other performance curve for classifier output - MATLAB perfcurve - MathWorks Nordic, [Online; accessed 9-March-2015].
URL http://se.mathworks.com/help/stats/perfcurve.html

[14] Lix — wikipedia, the free encyclopedia, [Online; accessed 10-November-2014] (2014).
URL http://en.wikipedia.org/wiki/LIX

[15] Wikipedia, List of latin words with english derivatives — wikipedia, the free encyclopedia, [Online; accessed 11-December-2014] (2014).
URL http://en.wikipedia.org/w/index.php?title=List_of_Latin_words_with_English_derivatives&oldid=635267509

[16] Wikipedia, List of english words of anglo-saxon origin — wikipedia, the free encyclopedia, [Online; accessed 11-December-2014] (2014).
URL http://en.wikipedia.org/w/index.php?title=List_of_English_words_of_Anglo-Saxon_origin&oldid=610864177

[17] D. Naber, Languagetool style and grammar check, [Online; accessed 12-February-2015] (2010).
URL https://languagetool.org/

[18] B. Spell, Java api for wordnet searching (jaws).
URL http://lyle.smu.edu/~tspell/jaws/

[19] Princeton University, About wordnet, [Online; accessed 11-December-2014] (2010).
URL http://wordnet.princeton.edu

[20] The stanford parser: A statistical parser, [Online; accessed 29-January-2015] (2015).
URL http://nlp.stanford.edu/software/lex-parser.shtml

[21] Stanford dependencies, [Online; accessed 29-January-2015] (2014).
URL http://nlp.stanford.edu/software/stanford-dependencies.shtml

[22] Wikipedia, Imperative mood — wikipedia, the free encyclopedia, [Online; accessed 25-February-2015] (2015).
URL http://en.wikipedia.org/w/index.php?title=Imperative_mood&oldid=648706828

[23] R. Nordquist, imperative sentence (grammar), [Online; accessed 2-February-2015]. URL http://grammar.about.com/od/il/g/impersent09.htm

[24] C. LoveToKnow, Examples of active and passive voice, [Online; accessed 30-January-2015]. URL http://examples.yourdictionary.com/examples-of-active-and-passive-voice.html#examples

[25] P. Norvig, How to write a spelling corrector, [Online; accessed 3-October-2014]. URL http://norvig.com/spell-correct.html

[26] M. Custard, T. Sumner, Using machine learning to support quality judgments, D-Lib Magazine 11 (10) 1082–9873.

[27] The university of pennsylvania(penn) treebank tag-set, [Online; accessed 15-December-2014]. URL http://www.comp.leeds.ac.uk/ccalas/tagsets/upenn.html

[28] Penn treebank ii tags, [Online; accessed 29-January-2015]. URL https://web.archive.org/web/20130517134339/http://bulba.sdsu.edu/jeanette/thesis/PennTags.html

[29] Stanford typed dependencies manual, [Online; accessed 19-March-2015]. URL http://nlp.stanford.edu/software/dependencies_manual.pdf

# Appendix A

# Digital Text Format

This format for digital texts is an `XML` based format inspired by `XHTML`. Except for the actual text, encoded with a subset of `XHTML`, the format also stores some meta information about the text, such as its name and grade.

## A.1 Format

The following tags are used for encoding the texts and their meta information:

### High level tags

`<document>` the root level tag.

`<meta>` contains the meta information about the text.

`<text>` contains the actual text.

### Meta tags

`<name>` an arbitrary name for the text.

`<grade>` the grade of the text.

### Markup tags

`<h1>, <h2>, <h3>, <h4>, <h5>, <h6>` contain headings at different levels.

`<p>` contains a paragraph of text.

`<ol>, <ul>, <li>` combined, they create ordered and unordered lists.

`<img>` represents an image at the place of the tag.

`<label>` represents an associated label for the `img` and `li` tags.

`<!-- -->` contains comments by the transcriber.

`<u>` shows that this text was underlined.

`<b>` shows that this text was bold.

`<i>` shows that this text was italic.

### A.1.1 Markup tags

All the markup tags have the same semantic meaning as in standard `XHTML` except for the `img` tag that is only used to indicate that there is an illustration. If the image has an associated label, that can be shown by including the `label` tag within the image tag. The `label` tag is also used to show the kind of bullet points in both ordered and unordered lists. The `u` tag can be used both in paragraphs and list items. The paragraphs can include the `img` tag for inline images.

### A.1.2 Lists

A list can be used both for structuring information and for enumerating items. But a list item can not contain text itself though it can contain `p` tags and images. Lists can also be nested.

### A.1.3 Lowercase sentence starts

The sign ¤ is used in place of the period when the following sentence starts with a lower case letter. The reason for this is to indicate that it is not an error in the digitization process but rather an error made by the author. If we would use the period in this case, this part of the text wouldn't be parsed as two separate sentences but rather as one sentence containing an abbreviation or acronym. After being used for sentence separation in the parser, the ¤ sign is converted to a period in the output.

## A.2 Example

```
<document>
  <meta>
    <name>B</name>
    <grade>4</grade>
  </meta>
  <text>
    <h1>Master thesis report</h1>
    <p>Hello, and welcome to the report...</p>
    <img><label>Figure 1</label></img>
  </text>
</document>
```

# Appendix B

# Stanford Parser Output

## B.1 The Penn Treebank Tag-set

### B.1.1 Word Level (Parts of Speech)

Source: *The University of Pennsylvania (Penn) Treebank Tag-set* [27]

| Tag | Description | Examples |
|---|---|---|
| $ | dollar | $ -$ –$ A$ C$ HK$ M$ NZ$ S$ U.S.$ US$ |
| “ | opening quotation mark | ‘ “ |
| ” | closing quotation mark | ’ ” |
| ( | opening parenthesis | ( [ { |
| ) | closing parenthesis | ) ] } |
| , | comma | , |
| – | dash | – |
| . | sentence terminator | . ! ? |
| : | colon or ellipsis | : ; ... |
| CC | conjunction, coordinating | & ’n and both but either et for less minus ... |
| CD | numeral, cardinal | mid-1890 nine-thirty forty-two one-tenth ... |
| DT | determiner | all an another any both del each either every ... |
| EX | existential there | there |
| FW | foreign word | gemeinschaft hund ich jeux habeas Haementeria ... |
| IN | preposition or conjunction, subordinating | astride among uppon whether out inside ... |
| JJ | adjective or numeral, ordinal | third ill-mannered pre-war regrettable oiled ... |

| | | |
|---|---|---|
| JJR | adjective, comparative | bleaker braver breezier briefer brighter brisker ... |
| JJS | adjective, superlative | calmest cheapest choicest classiest cleanest ... |
| LS | list item marker | A A. B B. C C. D E F First G H I J K One ... |
| MD | modal auxiliary | can cannot could couldn’t dare may might must ... |
| NN | noun, common, singular or mass | common-carrier cabbage knuckle-duster Casino ... |
| NNP | noun, proper, singular | Motown Venneboerger Czestochwa Ranzer ... |
| NNPS | noun, proper, plural | Americans Americas Amharas Amityvilles ... |
| NNS | noun, common, plural | undergraduates scotches bric-a-brac products ... |
| PDT | pre-determiner | all both half many quite such sure this |
| POS | genitive marker | ’ ’s |
| PRP | pronoun, personal | hers herself him himself hisself it itself me ... |
| PRP$ | pronoun, possessive | her his mine my our ours their thy your |
| RB | adverb | occasionally unabatingly maddeningly ... |
| RBR | adverb, comparative | further gloomier grander graver greater ... |
| RBS | adverb, superlative | best biggest bluntest earliest farthest first ... |
| RP | particle | aboard about across along apart around aside ... |
| SYM | symbol | % & ’ ” ”. ) ). * + ,. < = > @ A[fj] U.S ... |
| TO | ”to” as preposition or infinitive marker | to |
| UH | interjection | Goodbye Goody Gosh Wow Jeepers Jee-sus ... |
| VB | verb, base form | ask assemble assess assign assume atone avoid ... |
| VBD | verb, past tense | dipped pleaded swiped regummed soaked tidied ... |
| VBG | verb, present participle or gerund | telegraphing stirring focusing angering ... |
| VBN | verb, past participle | multihulled dilapidated aerosolized chaired ... |
| VBP | verb, present tense, not 3rd person singular | predominate wrap resort sue ... |
| VBZ | verb, present tense, 3rd person singular | bases reconstructs marks mixes ... |
| WDT | WH-determiner | that what whatever which whichever |
| WP | WH-pronoun | that what whatever whatsoever which who ... |
| WP$ | WH-pronoun, possessive | whose |
| WRB | Wh-adverb | how however whence whenever where whereby ... |

### B.1.2 Clause Level

Source: *Penn Treebank II Tags* [28]

| Tag | Description |
|---|---|
| S | simple declarative clause, i.e. one that is not introduced by a (possible empty) subordinating conjunction or a wh-word and that does not exhibit subject-verb inversion. |
| SBAR | Clause introduced by a (possibly empty) subordinating conjunction. |
| SBARQ | Direct question introduced by a wh-word or a wh-phrase. Indirect questions and relative clauses should be bracketed as SBAR, not SBARQ. |
| SINV | Inverted declarative sentence, i.e. one in which the subject follows the tensed verb or modal. |
| SQ | Inverted yes/no question, or main clause of a wh-question, following the wh-phrase in SBARQ. |

### B.1.3 Phrase Level

Source: *Penn Treebank II Tags* [28]

| Code | Description |
|---|---|
| ADJP | Adjective Phrase. |
| ADVP | Adverb Phrase. |
| CONJP | Conjunction Phrase. |
| FRAG | Fragment. |
| INTJ | Interjection. Corresponds approximately to the part-of-speech tag UH. |
| LST | List marker. Includes surrounding punctuation. |
| NAC | Not a Constituent; used to show the scope of certain prenominal modifiers within an NP. |
| NP | Noun Phrase. |
| NX | Used within certain complex NPs to mark the head of the NP. Corresponds very roughly to N-bar level but used quite differently. |
| PP | Prepositional Phrase. |
| PRN | Parenthetical. |

| | |
|---|---|
| PRT | Particle. Category for words that should be tagged RP. |
| QP | Quantifier Phrase (i.e. complex measure/amount phrase); used within NP. |
| RRC | Reduced Relative Clause. |
| UCP | Unlike Coordinated Phrase. |
| VP | Vereb Phrase. |
| WHADJP | Wh-adjective Phrase. Adjectival phrase containing a wh-adverb, as in how hot. |
| WHAVP | Wh-adverb Phrase. Introduces a clause with an NP gap. May be null (containing the 0 complementizer) or lexical, containing a wh-adverb such as how or why. |
| WHNP | Wh-noun Phrase. Introduces a clause with an NP gap. May be null (containing the 0 complementizer) or lexical, containing some wh-word, e.g. who, which book, whose daughter, none of which, or how many leopards. |
| WHPP | Wh-prepositional Phrase. Prepositional phrase containing a wh-noun phrase (such as of which or by whose authority) that either introduces a PP gap or is contained by a WHNP. |
| X | Unknown, uncertain, or unbracketable. X is often used for bracketing typos and in bracketing the...the-constructions. |

## B.2 Typed Dependencies

Source: *Stanford typed dependencies manual* [29]

| **Code** | **Full Name** |
|---|---|
| acomp | adjectival complement |
| advcl | adverbial clause modifier |
| advmod | adverb modifier |
| agent | agent |
| amod | adjectival modifier |
| appos | appositional modifier |
| aux | auxiliary |
| auxpass | passive auxiliary |
| cc | coordination |

| | |
|---|---|
| ccomp | clausal complement |
| conj | conjunct |
| cop | copula |
| csubj | clausal subject |
| csubjpass | clausal passive subject |
| dep | dependent |
| det | determiner |
| discourse | discourse element |
| dobj | direct object |
| expl | expletive |
| goeswith | goes with |
| iobj | indirect object |
| mark | marker |
| mwe | multi-word expression |
| neg | negation modifier |
| nn | noun compound modifier |
| npadvmod | noun phrase as adverbial modifier |
| nsubj | nominal subject |
| nsubjpass | passive nominal subject |
| num | numeric modifier |
| number | element of compound number |
| parataxis | parataxis |
| pcomp | prepositional complement |
| pobj | object of a preposition |
| poss | possession modifier |
| possessive | possessive modifier |
| preconj | preconjunct |
| predet | predeterminer |
| prep | prepositional modifier |
| prepc | prepositional clausal modifier |
| prt | phrasal verb particle |
| punct | punctuation |
| quantmod | quantifier phrase modifier |

| | |
|---|---|
| rcmod | relative clause modifier |
| ref | referent |
| root | root |
| tmod | temporal modifier |
| vmod | reduced non-finite verbal modifier |
| xcomp | open clausal complement |
| xsubj | controlling subject |

# Appendix C

# Evaluation Results

This appendix contains all the results from running the evaluation as described in Section 2.3 on the models defined in Section 3.2, except for the brute force models where we only present the top five models for each test. For each text set, algorithm, and performance measurement combination, we present this combination's best performance together with any varied parameters for the algorithm used to achieve this performance.

The values are presented in tables grouped into sections by the groups presented in Section 3.2. Both performance measurements share the same line in the tables, although the best varied parameters of the algorithms used are not necessarily the same for both results. When there are multiple models in each group, each model has a row in the tables and there are different tables for the different text set and algorithm combinations. When there is only a single model in the group, each text set and algorithm combination has a row in the same table.

The tables are sorted by descending AUC except for the brute force models, which are sorted by descending CCR. The reason for doing this is that calculating the AUC takes significantly longer time than calculating the CCR. Thus, we only calculate the AUC of the models that are in the top list. Even though we do not know if there are any models further down with higher AUC, we believe that it is interesting to present the AUC values for the best performing models measured by CCR. The results are commented on in Chapter 4.

## C.1 Single feature models

| **Instructional texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio | 0.67 | 33 | 0.72 | 25 |
| Verb Present Tense Verb Ratio | 0.71 | 7 | 0.71 | 8 |
| Imperative Sentence Ratio | 0.71 | 1 | 0.71 | 1 |
| Figures Per Word | 0.67 | 27 | 0.70 | 16 |

| | | | | |
|---|---|---|---|---|
| Lists Per Word | 0.73 | 1 | 0.69 | 1 |
| Paragraph Length In Characters Mean | 0.71 | 9 | 0.69 | 8 |
| Word Length In Characters Mean | 0.68 | 1 | 0.68 | 1 |
| Depth Items Per Word | 0.69 | 1 | 0.66 | 1 |
| Spelling Errors Per Word | 0.67 | 35 | 0.65 | 35 |
| STE Long Sentences Ratio Instructional | 0.67 | 25 | 0.65 | 1 |
| Active Voice Sentence Ratio | 0.67 | 25 | 0.65 | 32 |
| Word Depth At Bin Index | 0.67 | 34 | 0.65 | 35 |
| Paragraph Length In Words Variance | 0.67 | 33 | 0.65 | 10 |
| Paragraph Length In Characters Variance | 0.67 | 31 | 0.64 | 5 |
| Sentence Length In Words Mean | 0.67 | 35 | 0.64 | 1 |
| Paragraph Length In Sentences Variance | 0.67 | 33 | 0.63 | 31 |
| Sentence Length In Characters Variance | 0.67 | 25 | 0.63 | 18 |
| Ogden's Basic English Word Ratio | 0.67 | 33 | 0.62 | 7 |
| Paragraph Length In Sentences Mean | 0.67 | 31 | 0.62 | 33 |
| Verb Past Tense Verb Ratio | 0.67 | 29 | 0.62 | 6 |
| List Items Per Word | 0.67 | 31 | 0.62 | 35 |
| Sentence Length In Words Variance | 0.71 | 9 | 0.62 | 8 |
| Genitive Word Ratio | 0.67 | 9 | 0.60 | 26 |
| Depth Items At Depth Per Word In Whole | 0.67 | 27 | 0.60 | 2 |
| Paragraph Length In Words Mean | 0.70 | 5 | 0.60 | 11 |
| Sentence Length In Characters Mean | 0.67 | 27 | 0.59 | 36 |
| STE Approved Word Ratio | 0.67 | 33 | 0.58 | 2 |
| Figure Text Figure Ratio | 0.67 | 21 | 0.58 | 14 |
| Word Depth Ratios | 0.67 | 33 | 0.58 | 39 |
| Latin Etymology Word Ratio | 0.67 | 27 | 0.57 | 27 |
| STE Long Sentences Ratio Descriptive | 0.67 | 25 | 0.56 | 19 |
| List Lengths Variance | 0.67 | 31 | 0.56 | 9 |
| LIX Long Words Ratio | 0.67 | 26 | 0.56 | 38 |
| List Lengths Mean | 0.67 | 33 | 0.55 | 32 |
| Passive Voice Sentence Ratio | 0.67 | 21 | 0.54 | 37 |
| Word Length In Characters Variance | 0.67 | 33 | 0.52 | 7 |
| Other Errors Per Word | 0.67 | 21 | 0.52 | 1 |

| | | | | |
|---|---|---|---|---|
| STE Long Paragraph Ratio | 0.67 | 7 | 0.49 | 1 |
| Contraction Word Ratio | 0.67 | 13 | 0.49 | 16 |

| **Instructional texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word Depth At Bin Index | 0.67 | linear | 0.71 | linear |
| Figures Per Word | 0.67 | rbf | 0.68 | linear |
| Word Length In Characters Mean | 0.67 | linear | 0.65 | rbf |
| Verb Present Tense Verb Ratio | 0.66 | rbf | 0.63 | rbf |
| Anglo Saxon Etymology Word Ratio | 0.65 | linear | 0.62 | rbf |
| Depth Items Per Word | 0.66 | linear | 0.62 | rbf |
| Lists Per Word | 0.67 | linear | 0.62 | rbf |
| Ogden's Basic English Word Ratio | 0.67 | linear | 0.60 | rbf |
| Imperative Sentence Ratio | 0.67 | rbf | 0.58 | rbf |
| Passive Voice Sentence Ratio | 0.67 | rbf | 0.58 | rbf |
| STE Approved Word Ratio | 0.67 | linear | 0.58 | rbf |
| Paragraph Length In Words Mean | 0.67 | rbf | 0.57 | rbf |
| List Items Per Word | 0.67 | linear | 0.57 | rbf |
| Active Voice Sentence Ratio | 0.65 | linear | 0.55 | rbf |
| Sentence Length In Words Variance | 0.67 | linear | 0.55 | rbf |
| Sentence Length In Characters Variance | 0.67 | linear | 0.53 | rbf |
| Spelling Errors Per Word | 0.67 | rbf | 0.53 | rbf |
| Paragraph Length In Characters Mean | 0.67 | linear | 0.52 | rbf |
| STE Long Sentences Ratio Instructional | 0.67 | linear | 0.51 | rbf |
| Paragraph Length In Words Variance | 0.67 | rbf | 0.51 | linear |
| Paragraph Length In Characters Variance | 0.67 | rbf | 0.51 | rbf |
| STE Long Paragraph Ratio | 0.67 | rbf | 0.51 | linear |
| Contraction Word Ratio | 0.66 | linear | 0.51 | rbf |
| Genitive Word Ratio | 0.67 | rbf | 0.50 | linear |
| Word Length In Characters Variance | 0.67 | rbf | 0.50 | linear |
| List Lengths Variance | 0.67 | rbf | 0.50 | linear |
| Paragraph Length In Sentences Variance | 0.67 | linear | 0.50 | rbf |
| List Lengths Mean | 0.67 | rbf | 0.50 | rbf |

| | | | | |
|---|---|---|---|---|
| Other Errors Per Word | 0.67 | linear | 0.50 | linear |
| LIX Long Words Ratio | 0.67 | linear | 0.49 | linear |
| Figure Text Figure Ratio | 0.65 | rbf | 0.49 | linear |
| Depth Items At Depth Per Word In Whole | 0.67 | rbf | 0.49 | rbf |
| Paragraph Length In Sentences Mean | 0.67 | linear | 0.49 | linear |
| Verb Past Tense Verb Ratio | 0.67 | linear | 0.49 | linear |
| Latin Etymology Word Ratio | 0.67 | linear | 0.49 | linear |
| Sentence Length In Characters Mean | 0.67 | linear | 0.48 | linear |
| Sentence Length In Words Mean | 0.67 | rbf | 0.48 | linear |
| Word Depth Ratios | 0.63 | linear | 0.48 | linear |
| STE Long Sentences Ratio Descriptive | 0.67 | rbf | 0.48 | linear |

| **Instructional texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio | 0.84 | 9 | 0.92 | 9 |
| Lists Per Word | 0.72 | 7 | 0.81 | 6 |
| Depth Items Per Word | 0.80 | 1 | 0.81 | 1 |
| List Lengths Variance | 0.73 | 1 | 0.77 | 2 |
| Spelling Errors Per Word | 0.73 | 9 | 0.75 | 3 |
| Depth Items At Depth Per Word In Whole | 0.69 | 1 | 0.74 | 2 |
| Paragraph Length In Sentences Mean | 0.69 | 5 | 0.74 | 16 |
| Paragraph Length In Sentences Variance | 0.69 | 8 | 0.74 | 10 |
| Verb Present Tense Verb Ratio | 0.68 | 9 | 0.74 | 16 |
| Active Voice Sentence Ratio | 0.61 | 13 | 0.73 | 14 |
| List Items Per Word | 0.67 | 11 | 0.73 | 24 |
| Word Length In Characters Mean | 0.68 | 1 | 0.72 | 2 |
| Imperative Sentence Ratio | 0.63 | 15 | 0.71 | 15 |
| Paragraph Length In Characters Mean | 0.66 | 3 | 0.71 | 5 |
| Paragraph Length In Characters Variance | 0.61 | 5 | 0.71 | 21 |
| Word Depth At Bin Index | 0.68 | 3 | 0.69 | 20 |
| STE Approved Word Ratio | 0.68 | 3 | 0.68 | 5 |
| Paragraph Length In Words Variance | 0.66 | 3 | 0.68 | 16 |
| Latin Etymology Word Ratio | 0.61 | 9 | 0.67 | 11 |

| | | | | |
|---|---|---|---|---|
| List Lengths Mean | 0.65 | 13 | 0.67 | 22 |
| Paragraph Length In Words Mean | 0.63 | 1 | 0.67 | 5 |
| STE Long Sentences Ratio Descriptive | 0.66 | 1 | 0.66 | 1 |
| Ogden's Basic English Word Ratio | 0.61 | 3 | 0.63 | 2 |
| Sentence Length In Words Mean | 0.57 | 21 | 0.62 | 23 |
| Word Depth Ratios | 0.61 | 1 | 0.62 | 1 |
| Passive Voice Sentence Ratio | 0.63 | 3 | 0.60 | 19 |
| Figures Per Word | 0.59 | 7 | 0.60 | 5 |
| Genitive Word Ratio | 0.62 | 5 | 0.59 | 23 |
| Sentence Length In Words Variance | 0.56 | 21 | 0.58 | 12 |
| LIX Long Words Ratio | 0.55 | 23 | 0.58 | 21 |
| Sentence Length In Characters Mean | 0.57 | 21 | 0.57 | 24 |
| Word Length In Characters Variance | 0.56 | 23 | 0.56 | 5 |
| Verb Past Tense Verb Ratio | 0.55 | 23 | 0.53 | 3 |
| STE Long Sentences Ratio Instructional | 0.56 | 23 | 0.53 | 1 |
| Other Errors Per Word | 0.53 | 21 | 0.53 | 15 |
| Sentence Length In Characters Variance | 0.56 | 23 | 0.53 | 12 |
| Figure Text Figure Ratio | 0.53 | 23 | 0.50 | 26 |
| Contraction Word Ratio | 0.54 | 23 | 0.50 | 26 |
| STE Long Paragraph Ratio | 0.55 | 19 | 0.50 | 26 |

| **Instructional texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio | 0.83 | rbf | 0.93 | linear |
| Paragraph Length In Sentences Variance | 0.71 | rbf | 0.81 | rbf |
| Depth Items At Depth Per Word In Whole | 0.66 | linear | 0.79 | rbf |
| Lists Per Word | 0.70 | linear | 0.76 | rbf |
| Verb Present Tense Verb Ratio | 0.64 | rbf | 0.76 | linear |
| Depth Items Per Word | 0.68 | linear | 0.76 | linear |
| List Items Per Word | 0.67 | linear | 0.75 | linear |
| Paragraph Length In Sentences Mean | 0.64 | rbf | 0.75 | linear |
| Active Voice Sentence Ratio | 0.63 | linear | 0.75 | linear |
| Imperative Sentence Ratio | 0.64 | rbf | 0.73 | rbf |

| | | | | |
|---|---|---|---|---|
| Paragraph Length In Characters Mean | 0.60 | linear | 0.73 | linear |
| Spelling Errors Per Word | 0.75 | linear | 0.73 | rbf |
| STE Approved Word Ratio | 0.64 | linear | 0.72 | linear |
| Paragraph Length In Words Mean | 0.65 | linear | 0.71 | linear |
| Word Depth At Bin Index | 0.64 | linear | 0.69 | linear |
| List Lengths Mean | 0.67 | linear | 0.68 | rbf |
| Paragraph Length In Characters Variance | 0.65 | linear | 0.67 | linear |
| List Lengths Variance | 0.63 | rbf | 0.66 | rbf |
| Paragraph Length In Words Variance | 0.71 | rbf | 0.66 | linear |
| Passive Voice Sentence Ratio | 0.61 | rbf | 0.64 | rbf |
| Genitive Word Ratio | 0.63 | linear | 0.62 | linear |
| Sentence Length In Words Mean | 0.60 | linear | 0.62 | linear |
| Word Length In Characters Variance | 0.57 | rbf | 0.61 | rbf |
| Sentence Length In Characters Mean | 0.60 | linear | 0.61 | linear |
| Latin Etymology Word Ratio | 0.53 | rbf | 0.61 | rbf |
| Word Depth Ratios | 0.53 | linear | 0.60 | rbf |
| Word Length In Characters Mean | 0.53 | rbf | 0.60 | rbf |
| Sentence Length In Words Variance | 0.52 | rbf | 0.59 | rbf |
| Ogden's Basic English Word Ratio | 0.58 | linear | 0.58 | rbf |
| Other Errors Per Word | 0.54 | rbf | 0.55 | rbf |
| Figures Per Word | 0.47 | rbf | 0.55 | linear |
| Sentence Length In Characters Variance | 0.57 | linear | 0.53 | linear |
| STE Long Sentences Ratio Instructional | 0.60 | linear | 0.53 | linear |
| LIX Long Words Ratio | 0.49 | linear | 0.51 | rbf |
| STE Long Sentences Ratio Descriptive | 0.61 | rbf | 0.49 | rbf |
| STE Long Paragraph Ratio | 0.51 | rbf | 0.47 | rbf |
| Verb Past Tense Verb Ratio | 0.49 | linear | 0.44 | linear |
| Contraction Word Ratio | 0.52 | linear | 0.43 | linear |
| Figure Text Figure Ratio | 0.44 | rbf | 0.43 | linear |

| **Descriptive texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Active Voice Sentence Ratio | 0.75 | 7 | 0.73 | 24 |

| | | | | |
|---|---|---|---|---|
| LIX Long Words Ratio | 0.65 | 35 | 0.72 | 29 |
| Word Length In Characters Variance | 0.73 | 3 | 0.71 | 13 |
| Passive Voice Sentence Ratio | 0.73 | 5 | 0.69 | 5 |
| Paragraph Length In Characters Variance | 0.65 | 35 | 0.69 | 10 |
| Latin Etymology Word Ratio | 0.65 | 37 | 0.69 | 38 |
| Anglo Saxon Etymology Word Ratio | 0.73 | 1 | 0.68 | 1 |
| STE Approved Word Ratio | 0.71 | 1 | 0.68 | 1 |
| Sentence Length In Characters Variance | 0.65 | 35 | 0.68 | 14 |
| Verb Present Tense Verb Ratio | 0.65 | 35 | 0.65 | 31 |
| Verb Past Tense Verb Ratio | 0.65 | 7 | 0.64 | 36 |
| Paragraph Length In Words Variance | 0.65 | 35 | 0.64 | 34 |
| Depth Items At Depth Per Word In Whole | 0.65 | 35 | 0.62 | 32 |
| Sentence Length In Characters Mean | 0.65 | 33 | 0.62 | 32 |
| Word Length In Characters Mean | 0.65 | 35 | 0.62 | 38 |
| Sentence Length In Words Mean | 0.65 | 33 | 0.60 | 37 |
| Depth Items Per Word | 0.65 | 33 | 0.60 | 33 |
| Sentence Length In Words Variance | 0.65 | 35 | 0.58 | 1 |
| Word Depth Ratios | 0.65 | 13 | 0.57 | 39 |
| Spelling Errors Per Word | 0.65 | 31 | 0.57 | 39 |
| Imperative Sentence Ratio | 0.65 | 1 | 0.57 | 25 |
| Lists Per Word | 0.65 | 8 | 0.56 | 25 |
| Paragraph Length In Characters Mean | 0.65 | 23 | 0.56 | 39 |
| List Lengths Mean | 0.65 | 7 | 0.56 | 38 |
| List Items Per Word | 0.65 | 3 | 0.56 | 39 |
| Paragraph Length In Words Mean | 0.65 | 23 | 0.55 | 3 |
| STE Long Sentences Ratio Instructional | 0.65 | 31 | 0.55 | 17 |
| Paragraph Length In Sentences Variance | 0.65 | 23 | 0.55 | 37 |
| STE Long Sentences Ratio Descriptive | 0.65 | 29 | 0.53 | 1 |
| Paragraph Length In Sentences Mean | 0.66 | 15 | 0.53 | 1 |
| List Lengths Variance | 0.66 | 2 | 0.53 | 2 |
| Contraction Word Ratio | 0.65 | 9 | 0.53 | 39 |
| Genitive Word Ratio | 0.66 | 3 | 0.53 | 1 |
| Ogden’s Basic English Word Ratio | 0.65 | 17 | 0.52 | 36 |

| | | | | |
|---|---|---|---|---|
| Figures Per Word | 0.65 | 1 | 0.52 | 26 |
| Word Depth At Bin Index | 0.65 | 11 | 0.52 | 3 |
| Figure Text Figure Ratio | 0.65 | 1 | 0.52 | 25 |
| STE Long Paragraph Ratio | 0.65 | 15 | 0.51 | 36 |
| Other Errors Per Word | 0.65 | 27 | 0.49 | 39 |

| **Descriptive texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Active Voice Sentence Ratio | 0.69 | rbf | 0.74 | rbf |
| Word Length In Characters Variance | 0.62 | linear | 0.67 | rbf |
| LIX Long Words Ratio | 0.62 | linear | 0.60 | rbf |
| Lists Per Word | 0.64 | linear | 0.58 | rbf |
| Passive Voice Sentence Ratio | 0.65 | linear | 0.58 | rbf |
| STE Approved Word Ratio | 0.65 | linear | 0.57 | rbf |
| Verb Present Tense Verb Ratio | 0.65 | rbf | 0.56 | rbf |
| Latin Etymology Word Ratio | 0.61 | linear | 0.55 | rbf |
| STE Long Sentences Ratio Instructional | 0.65 | linear | 0.54 | rbf |
| Paragraph Length In Sentences Mean | 0.67 | linear | 0.54 | linear |
| Anglo Saxon Etymology Word Ratio | 0.65 | linear | 0.53 | rbf |
| Sentence Length In Characters Mean | 0.65 | linear | 0.53 | rbf |
| Verb Past Tense Verb Ratio | 0.65 | rbf | 0.53 | rbf |
| Depth Items At Depth Per Word In Whole | 0.64 | linear | 0.52 | linear |
| Word Length In Characters Mean | 0.65 | rbf | 0.52 | rbf |
| Sentence Length In Characters Variance | 0.65 | linear | 0.52 | rbf |
| Ogden's Basic English Word Ratio | 0.65 | linear | 0.52 | rbf |
| Depth Items Per Word | 0.65 | rbf | 0.52 | linear |
| Imperative Sentence Ratio | 0.65 | rbf | 0.51 | rbf |
| Genitive Word Ratio | 0.65 | rbf | 0.51 | rbf |
| Figures Per Word | 0.65 | rbf | 0.51 | rbf |
| Sentence Length In Words Variance | 0.65 | linear | 0.51 | linear |
| Figure Text Figure Ratio | 0.65 | rbf | 0.50 | rbf |
| List Lengths Mean | 0.65 | linear | 0.50 | linear |
| STE Long Paragraph Ratio | 0.65 | linear | 0.50 | linear |

| | | | | |
|---|---|---|---|---|
| Paragraph Length In Characters Variance | 0.65 | linear | 0.50 | linear |
| STE Long Sentences Ratio Descriptive | 0.65 | rbf | 0.50 | linear |
| Contraction Word Ratio | 0.65 | linear | 0.50 | linear |
| List Lengths Variance | 0.64 | rbf | 0.50 | linear |
| Other Errors Per Word | 0.65 | linear | 0.50 | linear |
| Word Depth Ratios | 0.62 | rbf | 0.50 | linear |
| Paragraph Length In Words Variance | 0.65 | linear | 0.49 | rbf |
| Paragraph Length In Sentences Variance | 0.65 | rbf | 0.49 | linear |
| Spelling Errors Per Word | 0.65 | rbf | 0.49 | linear |
| List Items Per Word | 0.65 | rbf | 0.49 | linear |
| Sentence Length In Words Mean | 0.65 | linear | 0.48 | linear |
| Word Depth At Bin Index | 0.62 | linear | 0.48 | rbf |
| Paragraph Length In Words Mean | 0.66 | linear | 0.46 | rbf |
| Paragraph Length In Characters Mean | 0.65 | linear | 0.45 | rbf |

| **Descriptive texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Active Voice Sentence Ratio | 0.68 | 9 | 0.72 | 16 |
| Paragraph Length In Characters Variance | 0.66 | 6 | 0.72 | 6 |
| Word Length In Characters Variance | 0.67 | 11 | 0.71 | 9 |
| Verb Present Tense Verb Ratio | 0.57 | 11 | 0.68 | 22 |
| Sentence Length In Characters Variance | 0.62 | 5 | 0.67 | 14 |
| Sentence Length In Characters Mean | 0.60 | 9 | 0.67 | 11 |
| Anglo Saxon Etymology Word Ratio | 0.66 | 1 | 0.66 | 1 |
| LIX Long Words Ratio | 0.59 | 13 | 0.66 | 10 |
| Latin Etymology Word Ratio | 0.59 | 23 | 0.66 | 21 |
| Paragraph Length In Words Variance | 0.63 | 3 | 0.65 | 8 |
| STE Approved Word Ratio | 0.67 | 5 | 0.65 | 7 |
| Sentence Length In Words Mean | 0.56 | 1 | 0.64 | 19 |
| Spelling Errors Per Word | 0.57 | 15 | 0.64 | 28 |
| Verb Past Tense Verb Ratio | 0.57 | 25 | 0.63 | 25 |
| Depth Items Per Word | 0.61 | 1 | 0.63 | 1 |
| Word Length In Characters Mean | 0.60 | 3 | 0.62 | 5 |

| | | | | |
|---|---|---|---|---|
| Passive Voice Sentence Ratio | 0.55 | 3 | 0.62 | 10 |
| Depth Items At Depth Per Word In Whole | 0.60 | 1 | 0.61 | 1 |
| Paragraph Length In Words Mean | 0.56 | 2 | 0.61 | 5 |
| Sentence Length In Words Variance | 0.55 | 1 | 0.59 | 17 |
| Paragraph Length In Characters Mean | 0.56 | 5 | 0.58 | 25 |
| Imperative Sentence Ratio | 0.50 | 1 | 0.58 | 28 |
| STE Long Sentences Ratio Descriptive | 0.53 | 13 | 0.57 | 28 |
| Word Depth Ratios | 0.54 | 1 | 0.57 | 28 |
| Paragraph Length In Sentences Variance | 0.55 | 5 | 0.55 | 26 |
| List Items Per Word | 0.50 | 11 | 0.55 | 26 |
| List Lengths Mean | 0.49 | 5 | 0.55 | 26 |
| Lists Per Word | 0.50 | 11 | 0.54 | 26 |
| Contraction Word Ratio | 0.49 | 9 | 0.54 | 28 |
| Word Depth At Bin Index | 0.54 | 1 | 0.54 | 1 |
| Ogden’s Basic English Word Ratio | 0.50 | 4 | 0.54 | 4 |
| STE Long Sentences Ratio Instructional | 0.52 | 1 | 0.53 | 1 |
| STE Long Paragraph Ratio | 0.47 | 14 | 0.52 | 8 |
| Other Errors Per Word | 0.41 | 26 | 0.52 | 26 |
| Genitive Word Ratio | 0.51 | 2 | 0.52 | 3 |
| Paragraph Length In Sentences Mean | 0.49 | 1 | 0.52 | 25 |
| List Lengths Variance | 0.50 | 2 | 0.52 | 2 |
| Figure Text Figure Ratio | 0.50 | 1 | 0.50 | 1 |
| Figures Per Word | 0.50 | 1 | 0.50 | 1 |

| **Descriptive texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Active Voice Sentence Ratio | 0.66 | rbf | 0.73 | linear |
| Verb Present Tense Verb Ratio | 0.58 | linear | 0.71 | linear |
| Sentence Length In Characters Mean | 0.56 | rbf | 0.68 | rbf |
| STE Approved Word Ratio | 0.56 | rbf | 0.68 | rbf |
| LIX Long Words Ratio | 0.59 | linear | 0.67 | linear |
| Word Length In Characters Variance | 0.59 | rbf | 0.67 | rbf |
| Sentence Length In Words Mean | 0.53 | rbf | 0.66 | rbf |

| | | | | |
|---|---|---|---|---|
| Paragraph Length In Characters Variance | 0.62 | rbf | 0.65 | linear |
| Anglo Saxon Etymology Word Ratio | 0.56 | linear | 0.64 | linear |
| Latin Etymology Word Ratio | 0.61 | linear | 0.64 | linear |
| Paragraph Length In Words Variance | 0.53 | rbf | 0.64 | linear |
| Sentence Length In Characters Variance | 0.54 | rbf | 0.64 | rbf |
| Depth Items Per Word | 0.48 | rbf | 0.63 | rbf |
| Spelling Errors Per Word | 0.58 | linear | 0.63 | linear |
| Verb Past Tense Verb Ratio | 0.62 | rbf | 0.62 | rbf |
| Passive Voice Sentence Ratio | 0.50 | linear | 0.61 | rbf |
| Depth Items At Depth Per Word In Whole | 0.56 | rbf | 0.60 | linear |
| Word Length In Characters Mean | 0.48 | rbf | 0.59 | rbf |
| Sentence Length In Words Variance | 0.50 | rbf | 0.57 | linear |
| Paragraph Length In Characters Mean | 0.42 | linear | 0.56 | rbf |
| Imperative Sentence Ratio | 0.47 | rbf | 0.55 | rbf |
| Paragraph Length In Sentences Mean | 0.42 | linear | 0.53 | linear |
| Paragraph Length In Words Mean | 0.42 | linear | 0.53 | rbf |
| Paragraph Length In Sentences Variance | 0.41 | rbf | 0.52 | linear |
| Word Depth Ratios | 0.43 | linear | 0.52 | linear |
| List Lengths Mean | 0.41 | linear | 0.52 | linear |
| Lists Per Word | 0.41 | rbf | 0.52 | rbf |
| STE Long Sentences Ratio Descriptive | 0.44 | linear | 0.51 | rbf |
| List Items Per Word | 0.42 | rbf | 0.50 | rbf |
| Figure Text Figure Ratio | 0.37 | rbf | 0.50 | rbf |
| Figures Per Word | 0.38 | rbf | 0.50 | linear |
| Word Depth At Bin Index | 0.39 | rbf | 0.50 | rbf |
| Ogden's Basic English Word Ratio | 0.40 | rbf | 0.49 | rbf |
| Other Errors Per Word | 0.36 | linear | 0.48 | linear |
| List Lengths Variance | 0.36 | rbf | 0.47 | rbf |
| Genitive Word Ratio | 0.35 | rbf | 0.47 | rbf |
| Contraction Word Ratio | 0.39 | linear | 0.46 | linear |
| STE Long Sentences Ratio Instructional | 0.34 | linear | 0.43 | linear |
| STE Long Paragraph Ratio | 0.34 | rbf | 0.42 | linear |

## C.2 Length features models

| **Instructional texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Simple means | 0.69 | 3 | 0.69 | 2 |
| Simple means and variances | 0.67 | 33 | 0.69 | 4 |
| Expanded variances | 0.67 | 27 | 0.64 | 11 |
| All means and variances | 0.67 | 32 | 0.64 | 6 |
| All length features | 0.67 | 33 | 0.62 | 5 |
| STE fixed limits advice instructional | 0.67 | 23 | 0.61 | 1 |
| Simple variances | 0.67 | 33 | 0.59 | 23 |
| Expanded means and variances | 0.67 | 31 | 0.58 | 35 |
| STE and LIX fixed limits advice instructional | 0.67 | 29 | 0.58 | 31 |
| Expanded means | 0.67 | 27 | 0.57 | 1 |
| STE and LIX fixed limits advice descriptive | 0.67 | 31 | 0.57 | 32 |
| STE fixed limits advice descriptive | 0.67 | 23 | 0.54 | 17 |

| **Instructional texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Simple means | 0.72 | rbf | 0.72 | rbf |
| Simple means and variances | 0.71 | rbf | 0.72 | rbf |
| All means and variances | 0.71 | rbf | 0.63 | rbf |
| All length features | 0.68 | rbf | 0.60 | rbf |
| Expanded variances | 0.67 | linear | 0.58 | linear |
| Simple variances | 0.67 | linear | 0.56 | rbf |
| STE fixed limits advice descriptive | 0.67 | rbf | 0.53 | rbf |
| STE fixed limits advice instructional | 0.67 | rbf | 0.52 | rbf |
| STE and LIX fixed limits advice descriptive | 0.66 | linear | 0.52 | linear |
| Expanded means and variances | 0.67 | linear | 0.49 | linear |
| STE and LIX fixed limits advice instructional | 0.66 | linear | 0.45 | linear |
| Expanded means | 0.67 | linear | 0.44 | rbf |

| **Instructional texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Simple variances | 0.66 | 3 | 0.71 | 6 |
| Expanded means | 0.69 | 1 | 0.68 | 1 |
| Simple means and variances | 0.70 | 3 | 0.67 | 3 |
| Expanded variances | 0.63 | 7 | 0.66 | 4 |
| All means and variances | 0.64 | 3 | 0.65 | 2 |
| All length features | 0.62 | 3 | 0.65 | 5 |
| Simple means | 0.63 | 1 | 0.63 | 1 |
| Expanded means and variances | 0.63 | 3 | 0.62 | 23 |
| STE and LIX fixed limits advice descriptive | 0.56 | 15 | 0.62 | 15 |
| STE fixed limits advice descriptive | 0.58 | 3 | 0.58 | 1 |
| STE and LIX fixed limits advice instructional | 0.57 | 7 | 0.56 | 7 |
| STE fixed limits advice instructional | 0.56 | 23 | 0.53 | 1 |

| **Instructional texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Simple means and variances | 0.61 | rbf | 0.76 | rbf |
| All means and variances | 0.55 | rbf | 0.74 | rbf |
| Simple variances | 0.62 | rbf | 0.73 | rbf |
| Simple means | 0.62 | rbf | 0.73 | rbf |
| All length features | 0.55 | linear | 0.70 | rbf |
| Expanded means | 0.60 | linear | 0.69 | linear |
| STE and LIX fixed limits advice instructional | 0.58 | rbf | 0.66 | rbf |
| Expanded variances | 0.63 | linear | 0.64 | linear |
| Expanded means and variances | 0.61 | linear | 0.61 | linear |
| STE and LIX fixed limits advice descriptive | 0.52 | linear | 0.56 | rbf |
| STE fixed limits advice instructional | 0.59 | linear | 0.52 | rbf |
| STE fixed limits advice descriptive | 0.54 | linear | 0.49 | linear |

| **Descriptive texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Simple variances | 0.71 | 13 | 0.74 | 13 |
| STE and LIX fixed limits advice instructional | 0.65 | 37 | 0.73 | 10 |
| All length features | 0.66 | 19 | 0.72 | 20 |
| Simple means and variances | 0.72 | 1 | 0.71 | 1 |
| Expanded variances | 0.69 | 3 | 0.70 | 7 |
| STE and LIX fixed limits advice descriptive | 0.65 | 37 | 0.69 | 28 |
| All means and variances | 0.69 | 1 | 0.68 | 23 |
| Expanded means and variances | 0.67 | 3 | 0.68 | 7 |
| Simple means | 0.65 | 31 | 0.65 | 30 |
| Expanded means | 0.65 | 29 | 0.62 | 34 |
| STE fixed limits advice instructional | 0.65 | 37 | 0.60 | 9 |
| STE fixed limits advice descriptive | 0.65 | 33 | 0.45 | 35 |

| **Descriptive texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Simple variances | 0.67 | rbf | 0.69 | rbf |
| Simple means and variances | 0.70 | rbf | 0.68 | rbf |
| STE and LIX fixed limits advice descriptive | 0.59 | linear | 0.67 | linear |
| All means and variances | 0.73 | rbf | 0.66 | rbf |
| Expanded means and variances | 0.66 | rbf | 0.66 | rbf |
| STE and LIX fixed limits advice instructional | 0.59 | linear | 0.65 | linear |
| All length features | 0.65 | rbf | 0.65 | rbf |
| STE fixed limits advice instructional | 0.65 | linear | 0.61 | rbf |
| Expanded variances | 0.63 | linear | 0.60 | linear |
| Expanded means | 0.68 | linear | 0.58 | linear |
| Simple means | 0.64 | linear | 0.57 | linear |
| STE fixed limits advice descriptive | 0.65 | linear | 0.49 | linear |

| **Descriptive texts, reduced set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Simple variances | 0.67 | 5 | 0.75 | 11 |
| Expanded means and variances | 0.69 | 3 | 0.70 | 3 |
| All means and variances | 0.63 | 3 | 0.70 | 22 |
| Expanded variances | 0.66 | 3 | 0.69 | 3 |
| Expanded means | 0.65 | 7 | 0.68 | 10 |
| STE and LIX fixed limits advice descriptive | 0.66 | 7 | 0.67 | 8 |
| All length features | 0.67 | 7 | 0.67 | 11 |
| Simple means and variances | 0.63 | 5 | 0.66 | 14 |
| Simple means | 0.57 | 3 | 0.64 | 4 |
| STE and LIX fixed limits advice instructional | 0.54 | 13 | 0.62 | 19 |
| STE fixed limits advice descriptive | 0.57 | 9 | 0.57 | 11 |
| STE fixed limits advice instructional | 0.42 | 1 | 0.50 | 31 |

| **Descriptive texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Expanded means | 0.61 | rbf | 0.68 | rbf |
| Simple variances | 0.63 | rbf | 0.67 | rbf |
| All length features | 0.51 | linear | 0.67 | rbf |
| Expanded means and variances | 0.64 | rbf | 0.67 | rbf |
| Expanded variances | 0.59 | rbf | 0.66 | linear |
| Simple means and variances | 0.57 | linear | 0.62 | linear |
| Simple means | 0.53 | linear | 0.62 | linear |
| STE and LIX fixed limits advice instructional | 0.54 | rbf | 0.60 | linear |
| STE and LIX fixed limits advice descriptive | 0.55 | linear | 0.60 | linear |
| All means and variances | 0.55 | linear | 0.60 | rbf |
| STE fixed limits advice descriptive | 0.52 | rbf | 0.57 | rbf |
| STE fixed limits advice instructional | 0.44 | rbf | 0.47 | rbf |

## C.3 Word etymology features model

| | **CCR** | **param.** | **AUC** | **param.** |
|---|---|---|---|---|
| Instructional texts, reduced set, SVM | 0.78 | rbf | 0.92 | linear |
| Instructional texts, reduced set, $k$NN | 0.79 | 7 | 0.89 | 8 |
| Instructional texts, complete set, $k$NN | 0.67 | 35 | 0.69 | 25 |
| Descriptive texts, complete set, $k$NN | 0.65 | 37 | 0.68 | 38 |
| Descriptive texts, reduced set, $k$NN | 0.59 | 17 | 0.68 | 27 |
| Descriptive texts, reduced set, SVM | 0.55 | linear | 0.63 | linear |
| Descriptive texts, complete set, SVM | 0.60 | linear | 0.59 | linear |
| Instructional texts, complete set, SVM | 0.63 | linear | 0.58 | linear |

## C.4 Basic English features model

| | **CCR** | **param.** | **AUC** | **param.** |
|---|---|---|---|---|
| Instructional texts, reduced set, SVM | 0.62 | linear | 0.70 | linear |
| Instructional texts, reduced set, $k$NN | 0.63 | 9 | 0.68 | 17 |
| Instructional texts, complete set, $k$NN | 0.67 | 35 | 0.59 | 12 |
| Descriptive texts, reduced set, SVM | 0.43 | rbf | 0.54 | rbf |
| Descriptive texts, complete set, $k$NN | 0.65 | 29 | 0.52 | 39 |
| Descriptive texts, reduced set, $k$NN | 0.47 | 5 | 0.50 | 31 |
| Instructional texts, complete set, SVM | 0.67 | linear | 0.49 | linear |
| Descriptive texts, complete set, SVM | 0.65 | linear | 0.45 | linear |

## C.5 Verb forms features models

| **Instructional texts, complete set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Verbs past and present tense | 0.67 | 34 | 0.72 | 39 |
| All verb forms features | 0.67 | 35 | 0.70 | 33 |
| Passive and active voice | 0.67 | 33 | 0.66 | 35 |

| **Instructional texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Verbs past and present tense | 0.64 | linear | 0.69 | linear |
| Passive and active voice | 0.65 | rbf | 0.66 | linear |
| All verb forms features | 0.65 | rbf | 0.66 | linear |

| **Instructional texts, reduced set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All verb forms features | 0.69 | 5 | 0.80 | 9 |
| Passive and active voice | 0.63 | 2 | 0.78 | 8 |
| Verbs past and present tense | 0.61 | 5 | 0.70 | 16 |

| **Instructional texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Passive and active voice | 0.65 | linear | 0.78 | linear |
| All verb forms features | 0.66 | linear | 0.73 | linear |
| Verbs past and present tense | 0.58 | linear | 0.71 | linear |

| **Descriptive texts, complete set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Passive and active voice | 0.70 | 3 | 0.71 | 19 |
| All verb forms features | 0.69 | 5 | 0.70 | 28 |
| Verbs past and present tense | 0.65 | 37 | 0.64 | 30 |

| **Descriptive texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Passive and active voice | 0.67 | rbf | 0.69 | linear |
| All verb forms features | 0.65 | rbf | 0.69 | linear |
| Verbs past and present tense | 0.66 | rbf | 0.67 | linear |

| **Descriptive texts, reduced set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Passive and active voice | 0.64 | 3 | 0.72 | 13 |
| Verbs past and present tense | 0.63 | 7 | 0.71 | 18 |
| All verb forms features | 0.64 | 7 | 0.70 | 7 |

| **Descriptive texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Verbs past and present tense | 0.64 | rbf | 0.73 | linear |
| All verb forms features | 0.62 | rbf | 0.71 | linear |
| Passive and active voice | 0.64 | rbf | 0.70 | linear |

## C.6 Grammar and spelling features models

| **Instructional texts, complete set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All grammar and spelling features | 0.67 | 34 | 0.63 | 12 |
| Spelling and other error features | 0.67 | 33 | 0.60 | 38 |
| Contraction and genitive features | 0.67 | 17 | 0.56 | 6 |

| **Instructional texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Spelling and other error features | 0.66 | linear | 0.64 | linear |
| All grammar and spelling features | 0.65 | rbf | 0.64 | linear |
| Contraction and genitive features | 0.68 | rbf | 0.53 | rbf |

| **Instructional texts, reduced set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All Grammar and spelling features | 0.67 | 3 | 0.70 | 7 |
| Spelling and other error features | 0.70 | 5 | 0.67 | 6 |
| Contraction and genitive features | 0.59 | 5 | 0.57 | 21 |

| **Instructional texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| All Grammar and spelling features | 0.72 | linear | 0.78 | rbf |
| Spelling and other error features | 0.73 | linear | 0.68 | rbf |
| Contraction and genitive features | 0.59 | linear | 0.56 | linear |

| **Descriptive texts, complete set, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All grammar and spelling features | 0.65 | 36 | 0.55 | 39 |
| Spelling and other error features | 0.65 | 27 | 0.52 | 39 |
| Contraction and genitive features | 0.65 | 15 | 0.51 | 38 |

| **Descriptive texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Spelling and other error features | 0.65 | linear | 0.51 | linear |
| Contraction and genitive features | 0.65 | rbf | 0.48 | linear |
| All grammar and spelling features | 0.64 | linear | 0.45 | linear |

| **Descriptive texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All Grammar and spelling features | 0.61 | 1 | 0.64 | 2 |
| Spelling and other error features | 0.56 | 1 | 0.59 | 19 |
| Contraction and genitive features | 0.52 | 1 | 0.55 | 2 |

| **Descriptive texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| All Grammar and spelling features | 0.58 | rbf | 0.69 | rbf |
| Spelling and other error features | 0.55 | linear | 0.58 | linear |
| Contraction and genitive features | 0.38 | rbf | 0.48 | rbf |

## C.7 Structural features models

| **Instructional texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Figure features | 0.67 | 29 | 0.66 | 10 |
| Word depth features | 0.67 | 35 | 0.63 | 33 |
| List features | 0.67 | 5 | 0.50 | 1 |
| All structural features | 0.67 | 5 | 0.50 | 1 |

| **Instructional texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word depth features | 0.67 | rbf | 0.71 | linear |
| All structural features | 0.67 | rbf | 0.66 | linear |
| Figure features | 0.67 | linear | 0.61 | linear |
| List features | 0.67 | linear | 0.57 | linear |

| **Instructional texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| List features | 0.67 | 1 | 0.71 | 23 |
| All structural features | 0.64 | 2 | 0.70 | 2 |
| Word depth features | 0.63 | 1 | 0.68 | 8 |
| Figure features | 0.58 | 2 | 0.62 | 2 |

| **Instructional texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word depth features | 0.70 | linear | 0.75 | linear |
| List features | 0.63 | rbf | 0.75 | rbf |
| All structural features | 0.63 | linear | 0.70 | linear |
| Figure features | 0.44 | rbf | 0.53 | rbf |

| **Descriptive texts, complete set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All structural features | 0.67 | 1 | 0.62 | 1 |
| Word depth features | 0.65 | 32 | 0.62 | 2 |
| List features | 0.66 | 2 | 0.56 | 25 |
| Figure features | 0.65 | 1 | 0.52 | 25 |

| **Descriptive texts, complete set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| List features | 0.65 | rbf | 0.52 | linear |
| Figure features | 0.65 | rbf | 0.51 | rbf |
| Word depth features | 0.62 | rbf | 0.47 | linear |
| All structural features | 0.60 | rbf | 0.47 | linear |

| **Descriptive texts, reduced set, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| All structural features | 0.65 | 1 | 0.65 | 1 |
| Word depth features | 0.64 | 1 | 0.65 | 1 |
| List features | 0.50 | 2 | 0.54 | 25 |
| Figure features | 0.50 | 1 | 0.50 | 1 |

| **Descriptive texts, reduced set, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word depth features | 0.46 | linear | 0.58 | rbf |
| All structural features | 0.48 | linear | 0.58 | linear |
| List features | 0.40 | linear | 0.51 | rbf |
| Figure features | 0.37 | rbf | 0.50 | rbf |

## C.8 All features model

|  | **CCR** | **param.** | **AUC** | **param.** |
|---|---|---|---|---|
| Instructional texts, reduced set, $k$NN | 0.68 | 3 | 0.75 | 18 |
| Instructional texts, complete set, $k$NN | 0.67 | 35 | 0.72 | 29 |
| Descriptive texts, complete set, SVM | 0.65 | rbf | 0.71 | rbf |
| Instructional texts, reduced set, SVM | 0.62 | linear | 0.71 | linear |
| Descriptive texts, complete set, $k$NN | 0.68 | 1 | 0.70 | 2 |
| Descriptive texts, reduced set, $k$NN | 0.63 | 3 | 0.66 | 3 |
| Instructional texts, complete set, SVM | 0.67 | rbf | 0.62 | linear |
| Descriptive texts, reduced set, SVM | 0.58 | linear | 0.62 | rbf |

## C.9 Brute force model selection

| **Instructional texts, complete set, combinations of 2 features, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Verb Present Tense Verb Ratio, Lists Per Word | 0.78 | 1 | 0.79 | 3 |
| Word Length In Characters Mean, Contraction Word Ratio | 0.77 | 1 | 0.76 | 1 |
| STE Long Sentences Ratio Instructional, Depth Items Per Word | 0.76 | 1 | 0.66 | 5 |
| Sentence Length In Characters Mean, Sentence Length In Words Mean | 0.75 | 1 | 0.73 | 1 |
| Verb Present Tense Verb Ratio, Figure Text Figure Ratio | 0.74 | 3 | 0.75 | 4 |
| *736 other models...* | - | - | - | - |

| **Instructional texts, complete set, combinations of 2 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word Length In Characters Mean, List Items Per Word | 0.76 | rbf | 0.75 | rbf |
| Word Length In Characters Mean, Depth Items Per Word | 0.75 | rbf | 0.74 | rbf |
| Anglo Saxon Etymology Word Ratio, Word Depth At Bin Index | 0.75 | linear | 0.81 | linear |
| Imperative Sentence Ratio, Active Voice Sentence Ratio | 0.75 | rbf | 0.62 | linear |
| Sentence Length In Characters Variance, Anglo Saxon Etymology Word Ratio | 0.74 | rbf | 0.67 | rbf |
| *736 other models...* | - | - | - | - |

| **Instructional texts, complete set, combinations of 3 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Word Length In Characters Mean, Passive Voice Sentence Ratio, Spelling Errors Per Word | 0.82 | 1 | 0.82 | 3 |
| Word Length In Characters Mean, Verb Present Tense Verb Ratio, Spelling Errors Per Word | 0.80 | 1 | 0.78 | 1 |
| Word Length In Characters Mean, Latin Etymology Word Ratio, Spelling Errors Per Word | 0.80 | 1 | 0.82 | 2 |
| Word Length In Characters Mean, Sentence Length In Words Mean, Spelling Errors Per Word | 0.78 | 1 | 0.83 | 3 |
| STE Long Paragraph Ratio, STE Long Sentences Ratio Instructional, Depth Items Per Word | 0.78 | 1 | 0.66 | 4 |
| *9134 other models...* | - | - | - | - |

| **Instructional texts, complete set, combinations of 3 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Word Length In Characters Mean, Sentence Length In Characters Mean, STE Approved Word Ratio | 0.83 | rbf | 0.80 | rbf |
| Word Length In Characters Mean, Sentence Length In Characters Mean, Spelling Errors Per Word | 0.83 | rbf | 0.87 | rbf |
| Word Length In Characters Mean, Sentence Length In Words Mean, STE Approved Word Ratio | 0.82 | rbf | 0.81 | rbf |
| Word Length In Characters Mean, Sentence Length In Words Mean, Spelling Errors Per Word | 0.82 | rbf | 0.87 | rbf |
| Word Length In Characters Mean, STE Approved Word Ratio, List Items Per Word | 0.80 | rbf | 0.78 | rbf |
| *9134 other models...* | - | - | - | - |

| **Instructional texts, reduced set, combinations of 2 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio | 0.85 | 7 | 0.93 | 13 |
| Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.84 | 7 | 0.94 | 15 |
| STE Long Paragraph Ratio, Anglo Saxon Etymology Word Ratio | 0.84 | 9 | 0.92 | 9 |
| Sentence Length In Words Mean, Anglo Saxon Etymology Word Ratio | 0.83 | 3 | 0.91 | 11 |
| LIX Long Words Ratio, Anglo Saxon Etymology Word Ratio | 0.83 | 3 | 0.93 | 16 |
| *736 other models...* | - | - | - | - |

| **Instructional texts, reduced set, combinations of 2 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.86 | rbf | 0.93 | linear |
| Word Length In Characters Mean, Anglo Saxon Etymology Word Ratio | 0.85 | linear | 0.92 | linear |
| LIX Long Words Ratio, Anglo Saxon Etymology Word Ratio | 0.84 | linear | 0.93 | linear |
| Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio | 0.83 | linear | 0.93 | linear |
| Word Length In Characters Variance, Anglo Saxon Etymology Word Ratio | 0.83 | rbf | 0.89 | linear |
| *736 other models...* | - | - | - | - |

| **Instructional texts, reduced set, combinations of 3 features, $k$NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio, Genitive Word Ratio, Spelling Errors Per Word | 0.89 | 1 | 0.96 | 14 |
| STE Long Sentences Ratio Instructional, Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.88 | 3 | 0.94 | 5 |
| Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio, Spelling Errors Per Word | 0.88 | 7 | 0.97 | 8 |
| Verb Present Tense Verb Ratio, Genitive Word Ratio, Spelling Errors Per Word | 0.88 | 3 | 0.89 | 10 |
| Sentence Length In Characters Mean, Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.87 | 3 | 0.94 | 12 |
| *9134 other models...* | - | - | - | - |

| **Instructional texts, reducet set, combinations of 3 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Verb Present Tense Verb Ratio, Genitive Word Ratio, Spelling Errors Per Word | 0.91 | linear | 0.94 | rbf |
| Sentence Length In Words Mean, Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.88 | rbf | 0.93 | rbf |
| STE Long Sentences Ratio Instructional, Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.87 | rbf | 0.94 | rbf |
| Anglo Saxon Etymology Word Ratio, Passive Voice Sentence Ratio, Spelling Errors Per Word | 0.87 | rbf | 0.95 | rbf |
| Sentence Length In Characters Mean, Anglo Saxon Etymology Word Ratio, Spelling Errors Per Word | 0.87 | rbf | 0.93 | rbf |
| *9134 other models...* | - | - | - | - |

| **Descriptive texts, complete set, combinations of 2 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Active Voice Sentence Ratio, List Lengths Mean | 0.78 | 7 | 0.72 | 23 |
| Active Voice Sentence Ratio, Lists Per Word | 0.77 | 7 | 0.74 | 16 |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio | 0.77 | 7 | 0.75 | 17 |
| Active Voice Sentence Ratio, List Items Per Word | 0.77 | 7 | 0.73 | 16 |
| Verb Past Tense Verb Ratio, Active Voice Sentence Ratio | 0.76 | 13 | 0.77 | 12 |
| *736 other models...* | - | - | - | - |

| **Descriptive texts, complete set, combinations of 2 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio | 0.76 | rbf | 0.76 | rbf |
| Active Voice Sentence Ratio, List Items Per Word | 0.74 | rbf | 0.76 | rbf |
| Active Voice Sentence Ratio, Lists Per Word | 0.73 | rbf | 0.78 | rbf |
| Paragraph Length In Words Variance, Active Voice Sentence Ratio | 0.73 | rbf | 0.75 | linear |
| Word Length In Characters Variance, Depth Items At Depth Per Word In Whole | 0.72 | rbf | 0.72 | rbf |
| *736 other models...* | - | - | - | - |

| **Descriptive texts, complete set, combinations of 3 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, List Lengths Mean | 0.81 | 5 | 0.77 | 4 |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Depth Items Per Word | 0.81 | 5 | 0.76 | 16 |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Lists Per Word | 0.81 | 5 | 0.76 | 12 |
| Sentence Length In Characters Variance, Paragraph Length In Words Mean, Depth Items Per Word | 0.81 | 1 | 0.76 | 1 |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, List Items Per Word | 0.80 | 5 | 0.75 | 4 |
| *9134 other models...* | - | - | - | - |

| **Descriptive texts, complete set, combinations of 3 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Depth Items At Depth Per Word In Whole | 0.81 | rbf | 0.79 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, List Items Per Word | 0.80 | rbf | 0.78 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Depth Items Per Word | 0.80 | rbf | 0.77 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Lists Per Word | 0.80 | rbf | 0.80 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, List Lengths Mean | 0.80 | rbf | 0.78 | rbf |
| *9134 other models...* | - | - | - | - |

| **Descriptive texts, reduced set, combinations of 2 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Word Length In Characters Variance, Ogden's Basic English Word Ratio | 0.76 | 1 | 0.77 | 1 |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio | 0.73 | 3 | 0.74 | 12 |
| Verb Past Tense Verb Ratio, Active Voice Sentence Ratio | 0.73 | 1 | 0.77 | 2 |
| Sentence Length In Words Variance, Verb Past Tense Verb Ratio | 0.72 | 1 | 0.72 | 1 |
| Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio | 0.72 | 3 | 0.78 | 5 |
| *736 other models...* | - | - | - | - |

| **Descriptive texts, reduced set, combinations of 2 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio | 0.74 | rbf | 0.76 | linear |
| Sentence Length In Characters Mean, Spelling Errors Per Word | 0.72 | rbf | 0.74 | rbf |
| Paragraph Length In Characters Variance, Imperative Sentence Ratio | 0.71 | rbf | 0.71 | linear |
| Active Voice Sentence Ratio, Depth Items At Depth Per Word In Whole | 0.71 | rbf | 0.73 | rbf |
| Paragraph Length In Characters Variance, Verb Present Tense Verb Ratio | 0.70 | rbf | 0.76 | rbf |
| *736 other models...* | - | - | - | - |

| **Descriptive texts, reduced set, combinations of 3 features, *k*NN** | **CCR** | **k** | **AUC** | **k** |
|---|---|---|---|---|
| Anglo Saxon Etymology Word Ratio, Verb Past Tense Verb Ratio, Active Voice Sentence Ratio | 0.79 | 5 | 0.83 | 4 |
| Paragraph Length In Words Variance, Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio | 0.78 | 3 | 0.79 | 3 |
| Paragraph Length In Characters Mean, STE Approved Word Ratio, Verb Present Tense Verb Ratio | 0.78 | 1 | 0.82 | 2 |
| Paragraph Length In Words Mean, STE Approved Word Ratio, Verb Present Tense Verb Ratio | 0.77 | 1 | 0.82 | 2 |
| Word Length In Characters Variance, Ogden's Basic English Word Ratio, List Lengths Variance | 0.77 | 1 | 0.78 | 1 |
| *9134 other models...* | - | - | - | - |

| **Descriptive texts, reduced set, combinations of 3 features, SVM** | **CCR** | **kernel** | **AUC** | **kernel** |
|---|---|---|---|---|
| Paragraph Length In Characters Variance, Anglo Saxon Etymology Word Ratio, Active Voice Sentence Ratio | 0.79 | linear | 0.87 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, List Items Per Word | 0.78 | rbf | 0.74 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Depth Items Per Word | 0.78 | rbf | 0.74 | rbf |
| Paragraph Length In Characters Variance, Active Voice Sentence Ratio, Lists Per Word | 0.78 | rbf | 0.75 | rbf |
| Paragraph Length In Characters Variance, Verb Present Tense Verb Ratio, Imperative Sentence Ratio | 0.77 | rbf | 0.76 | rbf |
| *9134 other models...* | - | - | - | - |