Evaluation of Features for Predicting Document Difficulty

by

Büşra Erdal

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfilment of
the requirements for the degree of
Master of Science

Sabancı University

July 2022

Acknowledgements

First of all, I would like to express my gratitude to my advisor Prof. Dr. Yücel Saygın for his valuable guidance, constant support, kind and encouraging personality through my studies. I am honored to be granted the chance to work with and to learn from him.

I owe a deep sense of gratitude to Stefan Räbiger for his great help with this work. His keen interest on me at every step of my research. He supported me in every decision and welcomed every mistake so that they never felt like failure. Without his contribution, this work would not have been completed.

I want to express my gratitude to my thesis jury members Assoc. Prof. Kemal Kılıç and Asst. Prof. Rahim Dehkharghani for their precious feedback to improve this thesis.

I am also thankful to my colleagues in the Audit Data Team at Garanti BBVA for their big support and belief in me, especially Hilmi Beydeş and İhsan Büyükuğur, who kindly gave me all the freedom needed during my studies, supported and encouraged me to start my master's degree.

I would like to thank my family for their support throughout my entire life. Their belief in me has kept my motivation high during this process. I would also like to thank my cat Pishi for her unconditional love, all the joy and emotional support.

I would like to thank all my friends, who stayed with me during my graduate studies. We were always together in this difficult process and we always gave each other motivation.

I also appreciate TUBITAK for funding me through the BİDEB-2210/A scholarship.

EVALUATION OF FEATURES FOR PREDICTING DOCUMENT DIFFICULTY

Büşra Erdal

Data Science, Master's Thesis, 2022

Thesis Supervisor: Yücel SAYGIN

Keywords: document difficulty, machine learning, explainable AI, conceptual complexity, readability assessment.

Abstract

Knowing the difficulty of a text document, in particular learning materials, has many benefits, such as recommending documents that are tailored towards a specific target group with the goal of maximizing understanding when reading these recommended documents.

While different factors exist that affect document difficulty, they capture different aspects of it. One of which is readability, which captures syntactical and lexical text properties and relates to linguistic difficulty. Another one is the background knowledge needed for readers to understand a given document because concepts therein might be more or less complex. Although both factors have been analyzed in isolation, their interplay is unknown. Similarly, the importance of both factors has not been examined, although addressing any of those problems could improve the understanding of document difficulty and thus pave the way towards more reliable models for predicting document difficulty.

Hence, this work investigates both problems by proposing a supervised model that extracts 20 features related to background knowledge and readability of a document to predict its difficulty. This model serves as the basis for analyzing the importance of these features and the interplay between background knowledge and readability for estimating document difficulty. We find that linguistic difficulty is more impor-

tant than background knowledge across all datasets. To the best of our knowledge, there are no datasets in the educational domain available for predicting document difficulty, thus we created one about biological concepts. We release this dataset to the research community in the hope to stimulate more research and provide more data to assess the reliability of methods for predicting document difficulty across different domains.

DOKÜMAN ZORLUĞUNU TAHMİN ETMEDE ÖZNİTELİKLERİN DEĞERLENDİRİLMESİ

Büşra Erdal

Veri Bilimi, Yüksek Lisans Tezi, 2022

Tez danışmanı: Yücel SAYGIN

Anahtar Kelimeler: doküman zorluğu, makine öğrenimi, açıklanabilir yapay zeka, kavramsal karmaşıklık, okunabilirlik analizi.

Özet

Bir metin belgesinin, özellikle eğitim materyallerinin zorluğunu bilmenin birçok faydası vardır. Bunlardan biri, okuduğunu anlamayı en üst düzeye çıkarmak amacıyla belirli bir hedef gruba yönelik uyarlanmış belgeler önermektir.

Doküman zorluğunu etkileyen farklı faktörler mevcut olmakla birlikte, bu faktörler doküman zorluğun farklı yönlerini yakalarlar. Bunlardan biri, sözdizimsel ve sözcüksel metin özelliklerini yakalayan ve dilbilimsel zorlukla ilgili olan okunabilirliktir. Bir diğeri, okuyucuların belirli bir dokümanı anlaması için gereken bilgi birikimidir, çünkü dokümandaki kavramlar okuyucu için karmaşık olabilir. Her iki faktör de ayrı ayrı analiz edilmiş olsa da, bu faktörlerin karşılıklı etkileşimleri bilinmemektedir. Benzer şekilde, bu faktörlerin doküman zorluğunu tahmin etmekteki önemi birlikte incelenmemiştir. Bu sorunlardan herhangi birinin ele alınması, doküman zorluğunun anlaşılmasını iyileştirebilir ve böylece doküman zorluğunu tahmin etmek için daha güvenilir modellerin yolunu açabilir.

Bu nedenle, bu çalışma, bir dokümanın zorluğunu tahmin etmek için gereken bilgi birikimi ve okunabilirliği ile ilgili 20 özniteliği çıkaran gözetimli bir model önererek her iki sorunu da araştırmaktadır. Bu model, doküman zorluğunu tahmin etmek için bu öznitelikleri önemini ve gereken birikim bilgisi ile okunabilirlik arasındaki karşılıklı etkileşimi analiz etmenin temelini oluşturur. Kullandığımız tüm veri kümelerinde

okunabilirliğin gereken bilgi birikiminden daha önemli olduğunu gözlemledik. Bildiğimiz kadarıyla, eğitim alanında belge zorluğunu tahmin etmek için mevcut bir veri seti yok, bu nedenle biyolojik kavramlar hakkında bir veri seti oluşturduk. Bu karşılaştırmalı veri setini, daha fazla araştırmayı teşvik etmek ve farklı alanlarda belge zorluklarını tahmin etmeye yönelik yöntemlerin güvenilirliğini değerlendirmek için daha fazla veri sağlamak umuduyla araştırma topluluğuna sunuyoruz.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The goal of this thesis is to understand how different features for supervised models contribute to the task of predicting document difficulty. Section 1.1 provides the context for this thesis by motivating why the task of predicting document difficulty is relevant, how current state-of-the-art methods address this task, and what the challenges are to improve results. Then Section 1.2 outlines how this thesis tackles these open problems and what the most important outcomes are.

1.1 Motivation

What if the difficulty level of learning materials was known? Learning materials could be tailored more accurately to the desired target audience. Similarly, more appropriate learning materials could be recommended on e-learning platforms such as [1] or search results rankings could be tailored to an individual's preferences. More generally speaking, knowing the difficulty level of any textual document, which learning materials are an instance of, provides benefits. For instance, documents could be specifically geared towards intellectually disabled individuals, which has been shown to improve their understanding [2]. Therefore, solving the problem of estimating document difficulty has attracted a lot of attention from researchers over time. For example, readability formulas like Dale-Chall [3] have been devised to measure the difficulty of texts by analyzing syntactical (sentence structure) and lexical (word difficulty) text properties. A plethora of methods [4, 5, 6, 7] have been proposed to address this problem known as automatic readability assessment with

the implicit assumption that once readers understand what they read, they can learn it. Learning materials are even assigned to different grade levels based on readability measures such as the Lexile reading level [8]. However, this is only one of many factors affecting difficulty simultaneously [9]. Automatic readability assessment suffers from the misconception that the ability to read a text is sufficient for understanding. For example, a textual document might confuse readers by having an incohesive train of thought, despite being written in simple language. Moreover, a document could mention different concepts, each of which requires the reader to have a varying degree of background knowledge to fully grasp the meaning of the text. This task, also known as predicting conceptual text complexity has been introduced only recently [10] in which the authors argue that document difficulty also depends on the required background knowledge to understand all concepts mentioned in the text. They propose different methods to address this task [10, 11, 12].

Thus, multiple factors contribute to the notion of document difficulty. However, there is no consensus on these factors when attempting to predict document difficulty automatically. Therefore, before diving into further challenges, we clarify what we mean by document difficulty in this work. Although working definitions of document difficulty have been proposed in the context of specific algorithms [13], they are too narrow and do not take into account findings from cognitive science. Hence we draw inspiration from theoretical frameworks that distinguish between external and internal factors of document difficulty. External factors include language proficiency (native speaker versus second language learner) [14] and motivation of a learner [15]. Internal factors include readability [5], the required background knowledge [10], and structural difficulty of the language, which can be measured by syntactical and morphological complexity [16]. In this work we limit ourselves to modeling document difficulty by means of internal factors since most external ones are hard to capture reliably.

While a broad range of factors affecting document difficulty have been explored in the literature, most of them have been only considered in isolation. Therefore it is an open question how important each of those factors is when combining them for estimating document difficulty. Likewise it is unclear how those factors interact.

An example of such an interaction is reported in [4], where the authors observed a recursive relationship between word and document difficulty: word difficulty correlates with the minimum difficulty of the document where the word occurs, and document difficulty correlates with the maximum word difficulty in that document. Acquiring such information for more features would be a step towards generating text that exhibits a desired difficulty level.

Another challenge for predicting document difficulty is the lack of appropriate datasets, especially in the educational domain. To the best of our knowledge, the only high-quality human-curated dataset for predicting document difficulty is Newsela [17], which contains news articles and simplified versions. Other datasets addressing this task are of low quality [17].

1.2 Overview of the Methodology and Contributions

In this thesis the overarching goal is to understand how different features affect document difficulty. To that end, we first define how we model document difficulty based on findings from cognitive science to avoid ambiguity when using the term "document difficulty". In particular, we consider two factors, namely *linguistic difficulty* and *inherent concept difficulty*. While linguistic difficulty covers lexical and syntactical properties of documents, inherent concept difficulty relates to the background knowledge required for understanding a document. In total, we extract 20 features for both factors together inspired by prior works. Features related to inherent concept difficulty are derived from a graph that is constructed on the basis of information obtained from an external knowledge base.

We leverage these features in our proposed supervised model to estimate the difficulty of textual documents. This model is used as a basis for investigating both the importance of the extracted features as well as their interplay for predicting document difficulty. To obtain more robust results, we conduct our experiments not only on Newsela, but also on a newly created dataset covering concepts taught in biology to both high school students and undergraduates in biology. To the best of

our knowledge, we provide the first dataset for the educational domain for predicting document difficulty. This new dataset also mitigates a drawback of Newsela, because Newsela was initially devised as a dataset to evaluate the simplification of sentences. Therefore, most sentences are aligned, which means that an easier variant of the sentence exists in the simplified document, whereas a harder variant of the sentence exists in the harder version of the same document. However, we show experimentally that this creates an artificial bias towards shorter sentences being simpler as removing and paraphrasing sentences are the main operations that were applied to simplify sentences. In contrast, the documents describing the same biological concepts in our newly created dataset stem from independent resources, thus sentences are unaligned. Hence we argue that our new dataset captures document difficulty more realistically.

Analyzing the importance of features and how their interplay affects predictions regarding document difficulty on both datasets reveals that features related to linguistic difficulty exert the biggest influence on the difficulty level of a document, while features gauging inherent concept difficulty affect the difficulty level mainly on our new dataset. In summary, our main contributions are:

1. We define document difficulty as a combination of linguistic difficulty, which is expressed by lexical and syntactical features, and inherent concept difficulty, which quantifies the background knowledge a reader needs to understand a document.

2. We propose a supervised method utilizing 20 features related to inherent concept and linguistic difficulty to estimate document difficulty.

3. To the best of our knowledge, we are the first to create a benchmark dataset for predicting document difficulty in the educational domain, specifically for biology.

4. We find that linguistic difficulty impacts document difficulty the most.

The rest of this thesis is structured as follows. Section 2 describes the most related work, covering the automatic estimation of document difficulty in terms of automatic readability assessment and predicting conceptual text complexity, and methods to

inspect and explain black-box models, which is known as explainable AI. Section 3 describes our proposed methodology for extracting the 20 features. This is followed by evaluating in Section 4 the performance of our supervised model and using this model to examine the impact of the features and their interplay. Last but not least, Section 5 sums up the thesis and explores avenues for future research.

Chapter 2

Related Work

In this section we review existing methods for predicting the difficulty of textual documents in Section 2.1 as well as methods for a closely related task, namely automatic readability assessment, in Section 2.2. This is followed by discussing popular methods for interpreting AI models in Section 2.3.

2.1 Predicting Document Difficulty

Most existing works predict the difficulty of specific aspects in a document, but not its overall difficulty. For instance, Huang et al. [18] predict the difficulty of questions by quantifying the difficulty contribution of each sentence in a document to each question by using an attention-based convolutional neural network. Similarly, in [19] question difficulty is estimated by leveraging student assessment data for multidimensional item response theory (IRT), which is an extension of IRT [20]. In [21] the authors adopt a similar approach based on IRT, but integrate semantic features derived from an external knowledge base to predict the difficulty of questions. Specifically, the authors introduce two features, namely Coherence and Specificity defined on the hierarchical structure of the knowledge base, which we adopt in this work.

The task of estimating the required background knowledge of textual descriptions to enable understanding of these texts has been introduced only recently and is referred to as predicting conceptual text complexity [10]. In the same paper the

authors suggest a supervised method capitalizing solely on 13 graph-based features from the DBpedia knowledge graph [22]. Since the feature extraction step is time-consuming and error-prone due to some missing in the graph, the same authors propose to combine those and an extended set of DBpedia-related features in [12], which improves their previous model. In [11] the authors propose an unsupervised method based on activation spreading over DBpedia subgraphs.

2.2 Automatic Readability Assessment

Automatic readability assessment aims to predict the appropriate grade level needed for being able to read a given text. The history of readability assessment is summarized in [23]. Initially, many hand-crafted readability formulas were proposed in the past that are still in use [23, 24], e.g., Dale-Chall [3]. These formulas typically estimate readability on the basis of a few features like sentence length or word difficulty. In [25] the authors analyzed the correlation of multiple readability formulas with two types of text genres, namely narrative and informative texts. They discovered that readability formulas performed well on informative texts. The authors of [26] further analyzed narrative and informative texts and found that many readability formulas underestimate the difficulty of literary texts, while overestimating the difficulty of informative texts because the latter contain less core vocabulary, which constitutes an important indicator of readability in readability formulas. Therefore, readability formulas are insufficient to estimate readability. These findings also motivate why more advanced features have been proposed and incorporated into more complex models over time. In [5] different feature categories to consider for readability assessment are summarized. An exhaustive list of 86 psycholinguistic features is described in [6], comprising features based on readability formulas, counts of part-of-speech tags, words based on specific word lists, Wordnet-based features, psycholinguistics, and parse trees for estimating the readability of texts. One disadvantage of these manually extracted features is that there might be other latent patterns in texts that have not been explored thoroughly. Thus, deep learning models that represent textual documents as high-dimensional vectors, also known as embeddings, might be promising, especially given the wealth of text data that has been collected over

decades of research on readability. One such method is presented in [27]. The authors fuse many of the features described in [6] with deep learning models, but find that the latter perform equally well without additional hand-crafted readability features. Another method employing deep learning is described in [4]. The authors present a semi-supervised approach to predict the discrete language levels of documents for second language learners. An advantage of their method is that it simultaneously predicts the difficulty/language level of words and documents using a graph convolutional network (GCN). To that end, the authors exploit the relationship between difficulty at the document and word level: first, word difficulty is correlated with the minimum difficulty of any document in which that word occurs. And second, document difficulty is correlated with the maximum difficulty of any word in that document. We incorporate the predicted language level from GCN as a feature in our proposed method for predicting document difficulty. Moreover, we employ GCN as a baseline in our evaluation.

While the task of automatic readability assessment [5, 7] is closely related to predicting document difficulty, it assumes that estimating the difficulty of reading a textual document is identical with its difficulty. However, text readability does not account for the required background knowledge to understand mentioned concepts in a document amongst other factors [9]. Thus, available data sets for readability assessment like [28, 29] are inappropriate for our evaluation. But in the remainder of this work we consider readability as one of the factors affecting document difficulty.

Our work differs from existing works in two aspects. First, unlike existing methods that regard document difficulty a result of predicting either the readability (automatic readability assessment) or required background knowledge (conceptual text complexity) of documents, we view document difficulty as the result of combining both factors. Secondly, our goal is to assess how different features contribute to document difficulty and how they interact. Furthermore, modeling document difficulty as a result of the factors readability and background knowledge enables us to analyze the importance of both factors for this classification task.

2.3 Model Explainability

Different methods exist to explain why black-box models predict a certain class label for a given document. One way to reveal insights about the inner workings of a black-box model is considering feature importance. The importance of features may be measured per document, which is known as local feature importance, or globally for the whole dataset resulting in a single ranking of feature importances, which is known as global feature importance. A popular method for measuring global feature importance is permutation importance [30]. Permutation importance is a popular method based on the idea of estimating the importance of feature f based on the following procedure. First a base model is trained on the original data. Then a second model is trained on the same dataset, but the values of f are permuted. The difference between the performances of the base model and the model trained on the permuted dataset indicates f's importance. Thus, negative permutation importance scores indicate that the second model with the permuted feature f performed better than the original model, which indicates that f is not important. However, it is known that permutation importance overestimates the importance of correlated features [31].

Diverse Counterfactual Explanations (DiCE) [32] pursues a different direction to allow interpreting black-box models. It generates counterfactuals, i.e. artificial documents, to answer the question: how much do the features of a document have to be adjusted for its class label to change? Since there are different possibilities of achieving the goal of switching a label, DiCE optimizes three objectives simultaneously, s.t. generated counterfactuals are diverse, feasible, and sparse. In this context feasibility means that one can select which of the features may be changed by DiCE to generate counterfactuals. For example, assuming that a person's age is among the features, then age should not be changed by DiCE in the counterfactual as a person cannot get younger in practice. Diversity implies that different combinations of features should change in different counterfactuals instead of updating mainly the same feature subset. Similarly, sparsity states that as few features as possible and as many features as necessary should be changed in counterfactuals. DiCE can also be employed to estimate feature importance motivated by the idea

that the more often a feature changes when generating counterfactuals for a given document, the more important that specific feature becomes. By generating counterfactuals for a single document, local feature importance can be estimated and summing these importance scores over all documents results in a global ranking of feature importances.

SHAP (SHapley Additive exPlanations) enables the interpretation of black-box models by measuring local and global feature importances [33]. This way SHAP can explain with local feature importance why a document has a certain predicted value and also how important each feature is in a dataset. SHAP is grounded in cooperative game theory and any feature importance computed by SHAP satisfies three desirable properties for an intuitive interpretation: local accuracy, consistency, and missingness. Local accuracy states that the sum of the importance scores of a document's features sum to its predicted value. Consistency states that if a model changes and a feature's value increases or stays the same, the feature's importance score also increases or stays the same. Missingness states that missing features have no contribution. It is known from cooperative game theory that only Shapley values satisfy these three properties. In SHAP a feature's importance score corresponds to its Shapley value. Shapley values are computed per document for local feature importance scores and are summed up to get a global ranking of the features over all documents.

At its core, SHAP assumes that the prediction of a trained and complex model f can be approximated by a simpler model g, s.t. the predicted label for document x, namely $f(x)$, is close to $g(x)$, i.e. $f(x) \approx g(x)$. A characteristic of g is that it is a linear additive model, which assigns an effect, i.e. an importance score or Shapley value, ϕ_i, to each feature. Summing up these effects results in $g(x)$, which approximates $f(x)$. Formally, this is expressed by:

$$f(x) = g(x') = \phi_0 + \sum_{i}^{n} \phi_i * x'_i \tag{2.1}$$

where x' represents the simplified input features, n denotes the number of simplified input features, and ϕ_0 indicates the prevalence of the positive class. The importance score or Shapley value ϕ_i of each feature x'_i is computed over all possible orderings, i.e. the power set, of the input features based on the idea that x_i's contribution is

measured as the difference between the model's performance with and without x_i. Instead of the actual feature values, simplified features indicate only the presence or absence of a feature in an ordering.

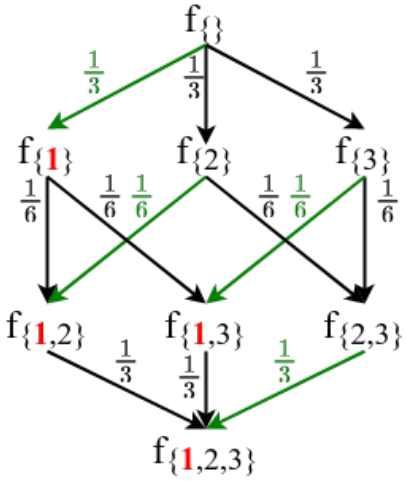


Figure 2.1: Intuition for computing the Shapley value for feature 1 in the feature set comprising features $\{1, 2, 3\}$: it is the sum of all marginal contributions indicated by green edges. Note that each row in the lattice sums to one.

The intuition behind Shapley values for a document x is depicted in Fig. 2.1. Given a feature set comprising features $\{1, 2, 3\}$, the importance of feature 1, which corresponds to its Shapley value, is computed as the sum of all marginal contributions of a model trained on a feature subset containing feature 1. This means that the difference in performance of a model trained on a feature subset containing feature 1 minus a model trained on the the same feature subset without feature 1 is attributed to the presence of feature 1. For example, the performance of a model trained on feature subset $\{1, 2\}$ minus the performance of a model trained on feature subset $\{2\}$ indicates the importance of feature 1. Summing up the contributions of feature 1 over all these marginal contributions highlighted with green edges yields exactly the Shapley value of feature 1 for document x. The weight of each marginal contribution, i.e. the weight of an edge in the lattice, sums up to one in every row to ensure that all rows contribute equally to the Shapley value.

Given the popularity of permutation importance, we utilize it in this thesis as a baseline method for estimating global feature importances. In addition, we also apply SHAP since it is based on game theory and it yields feature importances

that are more consistent with human intuition than other existing methods [33], we employ it not only for estimating feature importance, but also for examining the interplay among multiple features. Since SHAP already allows interpreting black-box models on a local and global level, we refrain from using DiCE.

Chapter 3

Problem Definition and Methodology

First and foremost, we formally define the problem setting in which our methodology operates and cover preliminaries in Section 3.1. Then Section 3.2 describes our methodology step by step.

3.1 Preliminaries and Problem Definition

In Section 3.1.1 we first formally state the problem that our proposed methodology addresses. It outlines the general workflow for training a supervised model that predicts document difficulty. Then Section 3.1.2 clarifies how we understand the term "document difficulty" and how we model it in this work. Last but not least, a substantial set of features for training the supervised model will be extracted from an external knowledge base, which is DBpedia in our case. Thus we briefly outline in Section 3.1.3 the basics about DBpedia that are utilized in the remainder of this thesis for feature extraction.

3.1.1 Problem Definition

Given a set of m unstructured text documents $D = \{d_1, d_2 \cdots d_m\}$, s.t. a set of k keywords $C_i = \{c_1, c_2, \cdots c_k\}$ is extracted by a keyword extraction algorithm for text document $d_i \in D$. These keywords represent concepts and thus we always refer to

them as concepts. Therefore, concept c_{ij} refers to the j-th concept of document d_i and C_i refers to the set of k concepts extracted from d_i. With the help of an entity linker each concept $c_{ij} \in C_i$ is linked to an entry in an external knowledge base to construct an undirected graph G_i. Features of d_i are extracted from G_i, the raw text of d_i, and the external knowledge base. Based on these features a supervised model M is trained to predict the difficulty of an unstructured text document d_i. Document difficulty is quantified by n discrete levels $L = 0, \dots n-1, n > 1$, where $L = 0$ refers to the most difficult description and $L = n-1$ refers to the easiest one.

3.1.2 Document Difficulty

Document difficulty informally describes how difficult it is for individuals to understand all ideas and arguments expressed in a given document. We model document difficulty as a combination of *linguistic difficulty* and *inherent concept difficulty*. We approximate linguistic difficulty by readability measures in line with [34]. Readability is defined as the overall effect of language usage and composition on the ability of readers to comprehend the document with ease [35]. Existing readability measures quantify readability as a certain combination of lexical features such as word difficulty and syntactical properties like sentence length. In line with [10], we consider conceptual complexity affecting document difficulty. The term "conceptual complexity" is defined in cognitive science and comprises conceptual primitives, which are to be thought of as building blocks. Humans either learn these primitives during childhood or they might even be innate and universal. Therefore, more complex concepts cover more conceptual primitives [36]. This notion bears similarities with knowledge space theory [37] that describes how humans learn new concepts. We refer to conceptual complexity as "inherent concept difficulty" throughout this thesis. It is measured as the cumulative background knowledge that a reader needs to know about all concepts mentioned in that document to understand it. Inherent concept difficulty is estimated based on features extracted from an external knowledge base, DBpedia [22] in our case, which allows us to exploit the correlation between the location of a concept in the knowledge graph and its difficulty level based on prior work.

3.1.3 DBpedia

DBpedia is a knowledge base for Wikipedia and has an entry for every Wikipedia article. In contrast to Wikipedia, DBpedia is stored as a directed graph adopting the RDF model [38]: each DBpedia entry is represented as a node and directed edges encode different types of relations that were extracted from Wikipedia articles. Therefore, DBpedia provides more structured access to the same data that Wikipedia provides, so accessing the information is easier. DBpedia uses SPARQL to retrieve desired subgraphs from the DBpedia knowledge graph. In this thesis we employ the following properties from DBpedia entries:

- dct:subject - contains the DBpedia categories in which that concept is used
- skos:broader - contains the parent categories of a category
- ^skos:broader - contains the child categories of a category
- dbo:abstract - contains the beginning of the corresponding Wikipedia article
- dbo:wikiPageWikiLink - contains the outgoing edges, i.e. all Wikipedia articles which the current DBpedia article links to
- ^dbo:wikiPageWikiLink - contains the incoming edges of a DBpedia entry, i.e. all Wikipedia articles that link to the current DBpedia entry

3.2 Methodology

Our methodology for training a supervised model for predicting document difficulty comprises three steps. First, we extract concepts from the given raw text documents and link (Section 3.2.1) them to the external knowledge base, which is DBpedia. In the second step, we construct an undirected graph per document (Section 3.2.2). Last but not least, we extract 20 features per document from different sources, namely from the raw text, from DBpedia and from the undirected graph constructed in the previous step.

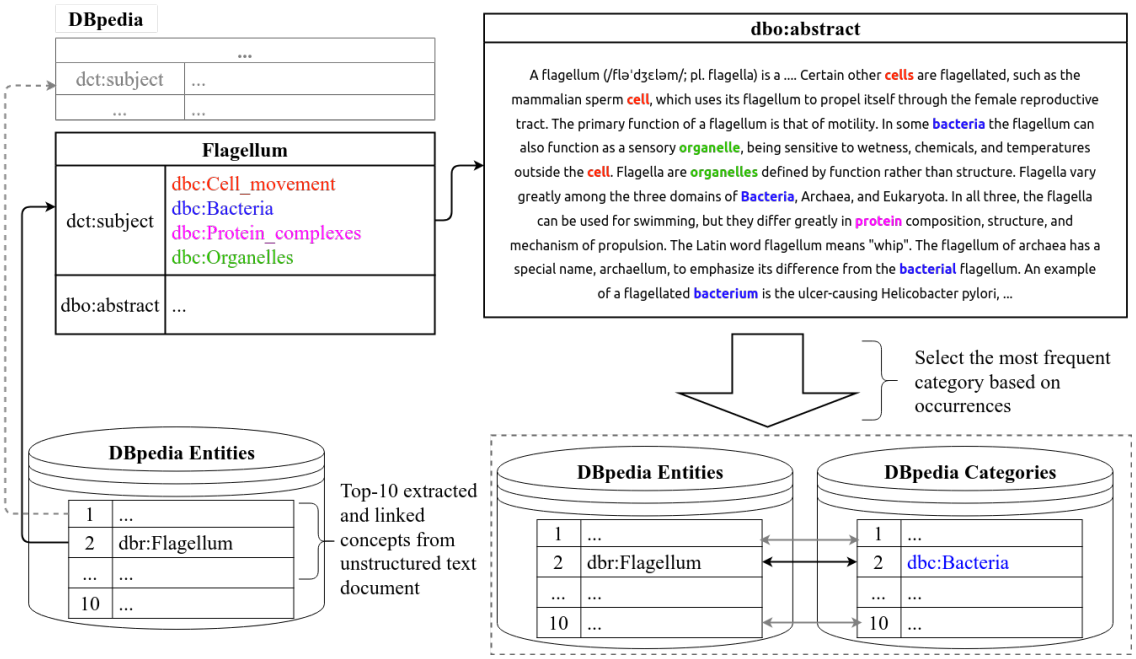


Figure 3.1: Illustration of the necessary steps to assign the given DBpedia entry "dbr:Flagellum" a single DBpedia category, which is "dbc:Bacteria".

3.2.1 Concept Extraction and Entity Linking

Some of the features we extract for an unstructured text document $d_i \in D$ are based on the concepts mentioned in d_i. Specifically, they exploit information from the DBpedia knowledge graph, but this requires linking the raw text to entries in the DBpedia knowledge graph. We identify concepts as follows. First, we extract the top-10 keywords from d_i with TopicRank [39] using PKE [40], which is an open source Python-based keyphrase extraction toolkit. By default 10 keywords are extracted as this has been shown empirically to be a good trade-off between actual keywords and noise. Each extracted keyword represents a concept. In the next step, these concepts are mapped to entries in the DBpedia knowledge graph with the help of DBpedia Spotlight [41]. The linking step involves disambiguating concepts, i.e. the concept "bank" might refer to the financial institution or a river bank depending on the context in d_i, so that the concept is mapped to the most likely DBpedia entry. To that end we employ DBpedia Spotlight [41].

Some of our features require DBpedia category (dct:subject) for a DBpedia entry. Since some entities belong to more than one category, we assign each DBpedia entry the most related category among its candidate categories. To that end we utilize

the frequency information of the candidate categories in the abstract (dbo:abstract) of the related DBpedia entry. To that end we count the number of occurrences of candidate categories and select the category with the highest number of occurrences. An example is given in Fig. 3.1, where the DBpedia entry "dbr:Flagellum" is assigned the single DBpedia category "dbc:Bacteria" as it occurs most frequently in the abstract of dbr:Flagellum.

3.2.2 Graph Construction

For each document $d_i \in D$ an undirected graph G_i is created based on the top-10 extracted concepts as follows. G_i is described in terms of its set of nodes V_i and its set of undirected edges E_i. Initially, Nodes in V_i correspond to the top-10 extracted concepts. For each of these concepts its corresponding DBpedia entry is retrieved and all concepts mentioned in its dbo:wikiPageWikiLink property are also added to V_i. An undirected edge between nodes $n_k \in V_i$ and $n_l \in V_i, n_k \neq n_l$ is added to E_i if both are connected via the dbo:wikiPageWikiLink property in DBpedia. As a result of this procedure, G_i might contain multiple disconnected components. Note that we only add additional nodes in V_i that are one hop (according to dbo:wikiPageWikiLink property) away from the top-10 extracted concepts. This decision is motivated by the findings in [42], in which the researchers found that relevant concepts lie close to each other, i.e., few hops apart, in the DBpedia knowledge graph.

3.2.3 Feature Extraction

Overall, our features can be grouped into three different categories as shown in Table 3.1.

Table 3.1: Short description of the extracted features representing a document d_i to predict its document difficulty. The set of top-10 concepts C_i was extracted with a keyword extraction algorithm and the undirected graph G_i with a set of nodes V_i was constructed for d_i. Note that we dropped "Average" from all feature names for readability.

Difficulty Factor	Source	Feature	Explanation
Inherent concept difficulty	DBpedia	Degree [21]	Average number of nodes that $c \in C_i$ is connected to in DBpedia.
		Specificity [21]	Average location of $c \in C_i$ in DBpedia
		Coherence [21]	Average semantic relatedness of $c \in C_i$ in DBpedia
		Support [41]	Average number of incoming edges of $c \in C_i$ in DBpedia
		Similarity [41]	Average probability that $c \in C_i$ refers to the linked DBpedia entry
		Rank2 [43]	Average confidence that $c \in C_i$ is disambiguated correctly and mapped to the corresponding DBpedia entry
		Offset	Confidence of the keyword extraction algorithm for $c \in C_i$ to represent a concept in d_i
	Graph	Clustering Coefficient [44]	Quantifies the extent to which the neighbors of $c \in C_i$ in G_i resemble a complete graph on average

Continued on next page

Table 3.1 – continued from previous page

Difficulty Factor	Source	Feature	Explanation
Inherent concept difficulty	Graph	PageRank [45]	Average relative importance of d_i as the average of the PageRank values of $c \in C_i$ in G_i
		Avg. Shortest Path	Average distance of $c \in C_i$ in G_i
		Nof Connected Components [46]	Number of connected subgraphs in G_i that are not part of any larger connected subgraph
		Global efficiency [47]	Average efficiency to send information concurrently in G_i across all pairs of nodes $n \in V_i$ in G_i
		Local efficiency [47]	Average resilience of G_i after removing $c \in V_i$ from G_i
		HITS Hubs [48]	Centrality of a concept $c \in C_i$ based on its outgoing edge in G_i
		HITS Authorities [48]	Centrality of a concept $c \in C_i$ based on its incoming edge in G_i
		Subgraph Centrality [49]	Number of closed walks of different length in G_i over $c \in C_i$ on average
		S-Metric [50]	Interconnectedness between nodes $c \in V_i$ with high node degree in G_i
		Degree Centrality	Average number of outgoing edges of $c \in C_i$ in G_i

Continued on next page

Table 3.1 – continued from previous page

Difficulty Factor	Source	Feature	Explanation
Linguistic difficulty	Raw text	Dale-Chall [3]	Lower scores indicate easier texts
		GCN [4]	Predicts the CEFR level (A1, ... C2) of d_i

DBpedia-based Features

DBpedia-based features are related to inherent concept difficulty as they encode structural information from the DBpedia knowledge graph by capturing how densely connected a concept is, how it relates to similar concepts, and where exactly that concept is located in the graph. All features from this category are computed on the DBpedia knowledge graph.

Average Degree(d_i), the Average Node Degree or Popularity [21] of document d_i, indicates how well d_i is connected to other concepts in the DBpedia knowledge graph on average:

$$\text{Average Degree}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{in-degree}(c) + \text{out-degree}(c) \tag{3.1}$$

where, out-degree(c) refers to the number of outgoing edges from c and in-degree(c) denotes the number of incoming edges of c based on the dbo:wikiPageWikiLink property in DBpedia. *We hypothesize that higher Average Degree is indicative of easier documents, because due to more connections in d_i, the concepts $c \in C_i$ are more likely to be familiar.*

Average Specificity(d_i) [21] of document d_i indicates the average location of d_i in the DBpedia hierarchy:

$$\text{Average Specificity}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Specificity(c)depth of } c\text{'s category from the root category} \tag{3.2}$$

where $(Specificity)(c)$ [21] of concept $c \in C_i$ is measured in the DBpedia knowledge graph as the distance of c's DBpedia category (dct:subject) from the root cat-

egory. *We hypothesize that for understanding documents with higher Average Specificity more knowledge is required, thus higher values of specificity result in harder documents.*

Average Coherence(d_i) of document d_i indicates how similar the concepts $c \in C_i$ are on average based on the DBpedia knowledge graph structure:

$$\text{Average Coherence}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Coherence}(c, C_i) \tag{3.3}$$

where Coherence(c) [21] of concept c indicates how similar c is on average in terms of related categories (skos:broader) to all the other concepts in C_i in the DBpedia knowledge graph. *We hypothesize that higher Average Coherence value indicate more coherent concepts, which reduces the mental load of readers trying to understand such documents. And this, in turn, makes such documents easier to understand.*

DBpedia Spotlight utilizes features for entity linking in a generative entity mention model [51] that produces for each concept $c \in C_i$ in document d_i's raw text a score how likely c refers to a DBpedia entity. For disambiguation this model utilizes different features describing the probability distribution of keyword occurrences in the text of d_i compared with the distribution over all DBpedia entities. We incorporate three of these features provided by DBpedia Spotlight which are called Support, Similarity, Rank2. In addition, we also compute a fourth feature based on DBpedia's Offset feature to which we also refer as Offset.

Average Support(d_i) of document d_i describes the average number of incoming edges in the DBpedia knowledge graph:

$$\text{Average Support}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Support}(c) \tag{3.4}$$

where Support(c) [41] describes the number of incoming edges of concept $c \in C_i$. It is used by DBpedia Spotlight as the minimum threshold c has to exceed for it to be linked to entities. *We hypothesize that higher Average Support values correlate with lower document difficulty because they indicate that concepts are more common, which makes them more likely to be understood by readers.*

Average Similarity(d_i) of document d_i refers to the average probability that concepts

in d_i link to their matched DBpedia entries:

$$\text{Average Similarity}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Similarity}(c) \tag{3.5}$$

where Similarity(c) [41] represents the average probability that a concept c from d_i links to the matched DBpedia entry. *We hypothesize that if the disambiguation of c is easy for DBpedia, it is also easy for humans. Thus, high Average Similarity values correlate with easier documents.*

Average Rank2(d_i) of document d_i estimates the average difference between the concepts' matched DBpedia entries and their second best DBpedia entry alternatives:

$$\text{Average Rank2}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} Rank2(c) \tag{3.6}$$

where Rank2(c) [43], the average percentage of second rank for concept c, which corresponds to the Similarity($\bar{c}$ of the next best candidate $\bar{c}$ for a concept c in d_i's text compared to Similarity(c). *We hypothesize that higher differences increase confidence in the disambiguation and therefore higher values of Average Rank2 characterize easier documents.*

Average Offset(d_i) of document d_i represents the average confidence that the keyword extraction algorithm identified a concept that has an entry in DBpedia:

$$\text{Average Offset}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Offset}(c) \tag{3.7}$$

where Offset(c) represents the confidence of the keyword extraction algorithm for concept candidate c to represent an actual concept in d_i. *We hypothesize that higher confidence scores correlate with easier documents, because identifying concepts helps also a learner to see relationships across concepts.*

Graph-based Features

Graph-based features are extracted for each document d_i from G_i, which is constructed according to Section 3.2.2.

Average PageRank(d_i) of document d_i describes the average importance of its concepts in G_i:

$$\text{Average PageRank}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{PageRank}(c) \tag{3.8}$$

where PageRank(c) [45], of concept c measures the relative importance of c in G. *We hypothesize that high PageRank values indicate that d_i is easier because many other concepts link to $c \in C_i$, thus c is more common on average.*

Similar to Average PageRank(d_i), we rank concepts $c \in C_i$ in document d_i based on HITS [48] in G_i, where we distinguish between Average HITS Hubs(d_i) and Average HITS Authorities(d_i). The former measures the centrality of a concept $c \in C_i$ based on its outgoing edges in G_i, whereas the latter utilizes c's incoming edges. *We hypothesize that both high Average HITS Authorities values and high Average HITS Hubs values, are more likely to occur in easier documents because such documents contain highly connected concepts that learners might be already familiar with.*

Average Shortest Path(d_i) of document d_i measures the average of shortest paths among $c \in C_i$ in G_i:

$$\text{Average Shortest Path}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Shortest Path}(c) \tag{3.9}$$

where Shortest Path(c) measures measures the average of shortest paths from $c \in C_i$ to $c_j \in C_i - \{c\}$. *We hypothesize that high Average Shortest Paths indicate that d_i mentions many unrelated or specialized concepts, which makes understanding d_i harder.*

$AverageClusteringCoefficient(d_i)$ of document d_i describes how closely connected G_i is on average:

$$\text{Average Clustering Coefficient}(d_i) = \frac{1}{|C_i|} \sum_{c \in V_i} \text{Clustering Coefficient}(c) \tag{3.10}$$

where Clustering Coefficient(c) [44], of concept $c \in C_i$ in G_i describes how closely connected c's neighbors are in G_i. If the neighbours of c are connected among themselves, then the clustering coefficient of c will be higher. Therefore, the nodes with low clustering coefficient tend to be more general. *We hypothesize that lower Average Clustering Coefficient values correlate with easier documents, because with lower Average Clustering Coefficient values a document contains more general nodes which makes understanding it easier.*

S-Metric(d_i) of document d_i is a measure of interconnectedness between hub nodes

(nodes with very high degree) of concepts $c \in V_i$ in G_i [50]. *We hypothesis that high S-Metric values make documents easier, because more interconnected concepts are likely more familiar, which enhances text understanding.*

Average Degree Centrality(d_i) of document d_i measures how connected concepts in d_i are:

$$\text{Average Degree Centrality}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Degree Centrality}(c) \tag{3.11}$$

where Degree Centrality(c) describes the number of edges of concept $c \in C_i$ in G_i. *We hypothesize that higher values of Average Degree Centrality correlate with easier documents because well-connected concepts increase the chances of familiarity with them once readers come across them.*

Average Subgraph Centrality(d_i) of document d_i describes how many closed walks of different length exist in G_i on average:

$$\text{Average Subgraph Centrality}(d_i) = \frac{1}{|C_i|} \sum_{c \in C_i} \text{Subgraph Centrality}(c) \tag{3.12}$$

where Subgraph Centrality(c) of a concept $c \in C_i$ is the sum of closed walks of different lengths in G_i starting and ending at c [49]. *We hypothesize that higher Average Subgraph Centrality values correlate with easier documents, because fewer concepts need to be understood.*

Global Efficiency(d_i) [47] of document d_i is a measure of how efficiently G_i exchanges information concurrently [47]. *We hypothesize that higher Global Efficiency values indicate easier documents because high efficiency implies short paths in the graph, which makes concepts*

Local Efficiency(d_i) of a document d_i measures, on average, how efficiently G_i can send information concurrently once all $c \in C_i$ are removed [47]. Thus, high Local Efficiency values indicate that a document is more robust. *We hypothesize that high Local Efficiency values are more likely to occur in easier documents because the same ideas are still expressed by neighboring concepts, which could be included in the documents.*

Nof Connected Components(d_i), the number of (Nof) connected components [46], of document d_i counts the number of disjoint sets in G_i, and there is a path among

all nodes within a disjoint set, but no path to other disjoint sets. *We hypothesize that lower values of Nof Connected Components will correlate with easier documents because having fewer disjoint sets suggests that concepts are more related.*

Features Related to Language Level and Readability

Dale-Chall readability score [3] distinguishes six levels, where the lowest level denotes the easiest text. Instead of using the categories, we directly utilize the raw scores. $DC(d_i)$, the Dale-Chall readability score of document d_i's raw text, is computed based on:

$$DC(d_i) = 0.1579\left(\frac{\#\text{difficult words}}{\#\text{words}} * 100\right) + 0.0496\left(\frac{\#\text{words}}{\#\text{sentences}}\right) \tag{3.13}$$

In addition, we utilize GCN [4], which outputs in the classification setting for each document a discrete language level according to the Common European Framework of Reference (CEFR) [52], i.e., one level from $\{\text{A1}, \text{A2}, \text{B1}, \text{B2}, \text{C1}, \text{C2}\}$). Note that GCN is trained in a transductive setting, meaning that labeled and unlabeled concepts and words are included in the graph. As labeled resources we utilize only the two publicly available data sets provided by the authors, which are Cambridge [53] and CEFR-J [54]. Similarly, for training we rely on the recommended configuration for the classification setting: $\alpha = 0.3$, learning rate $= 0.0005$, dropout probability $= 0.5$, number of hidden units $= 512$, number of hidden layers $= 2$, epochs $= 500$, PMI window width $= 5$. GCN takes as input the raw, unprocessed documents.

By convention, we will drop *Average* from all feature names hereafter for the sake of clarity, because whenever we refer to a feature, we refer to the averaged version defined for documents.

Chapter 4

Evaluation

To understand how different features contribute to the prediction of document difficulty, we focus in our experiments on three specific research questions (RQs). To gauge the capabilities of our model, we first quantify its performance for predicting document difficulty and analyze how well does it fares against other methods (RQ1). To better understand the contribution of each extracted feature for the classification task, we further analyze in RQ2 the feature importance of our model trained for RQ1. Last but not least, in RQ3 we aim to explore more detailed relationships among the features in the hope to identify common patterns that generalize across datasets.

4.1 Datasets

In our experiments, we use two different data sets, namely Newsela [17] and Biology, where we created the latter one. Newsela is comprised of 1905 newspaper articles with four more and more simplified versions that were created by professional editors according to the Lexile readability measure [55]. Thus, this is a high-quality corpus containing five versions of the same news articles with difficulty labels from 0 (hardest) to 4 (easiest), where each news article represents a separate document. One shortcoming of Newsela is that it was created mainly for sentence simplification, i.e. the majority of sentences from the original article have a simplified version, which allows aligning those sentences across all difficulty levels and predict their difficulty. This is an unrealistic way of creating simpler versions of the original article. Ideally,

each article would have been rewritten from scratch to have a more realistic dataset for estimating conceptual complexity. Despite this known weakness, Newsela is used in related works due to the lack of alternatives [10, 11, 12]. For example, Simple Wikipedia [56] is known to be of low quality, which is the reason why it is not used anymore.

To address the drawback of Newsela and to overcome the lack of available datasets for predicting conceptual complexity, especially for the educational domain, we created a new dataset for the domain of biology to which we simply refer as Bio in the remainder. It is based on two independently written open source biology books, one tailored to high school students (HIGHSCHOOL)[1] and one tailored to students with a biology major (UNIVERSITY)[2]. This way concepts are described independently, which resembles a more realistic setting for measuring conceptual text complexity than Newsela as sentences are not aligned. To identify concepts that are explained in both books, a domain expert matched concepts manually. If concepts from HIGHSCHOOL were explained in UNIVERSITY, it was counted as a match and the respective text sections were extracted into documents. Difficulty labels were assigned to these documents automatically based on the resource: texts from HIGHSCHOOL were regarded easy, whereas the corresponding explanations from UNIVERSITY were considered hard. With this methodology, a total of 174 concepts was identified that is explained in both books. Thus, Bio comprises 174 concepts with binary difficulty labels 0 (hard) and 1 (easy).

4.2 Experimental Design

To address RQ1, we train multiple models using our extracted features. In line with Section 3.1.1, our goal is to predict the difficulty level of document d_i. Valid labels are the n discrete levels $L = 0 \dots n-1$, where 0 refers to the hardest description and $n-1$ to the easiest one. For Newsela, we are given $n = 5$ versions of the same news article, and each one represents a separate document. In contrast, for Bio we have

[1] https://ia600307.us.archive.org/22/items/ost-biology-ck_12_biology_i/CK_12_Biology_I.pdf

[2] https://assets.openstax.org/oscms-prodcms/media/documents/Biology2e-WEB_ICOFkGu.pdf

$n = 2$ documents describing the same biological concept. Our goal is to predict the correct difficulty level of a given document d_i.

To address RQ2, we estimate the feature importances of the previously trained model from RQ1 according to different methods as each one yields different rankings of features. With multiple methods we can verify the consistency of the obtained rankings. Specifically, we rely on widely applicable methods that do not make any assumptions about a trained model, because they consider it as a black box model. In our case, we use permutation importance [30] and SHAP [33], which are both explained in detail in Section 2.3, to measure global feature importances, i.e. importance of the features over the whole datasets.

To address RQ3, we investigate features of our trained model as follows. Since global feature importance alone potentially misses subtler effects, we also utilize SHAP for estimating local feature importance, i.e. the importance of each feature for the prediction of single documents. Moreover, we examine potential interactions among features and last but not least, we also repeat these two analyses separately for correctly predicted documents and for incorrectly predicted documents to identify differences.

4.3 Visual Interpretations of SHAP Values

For understanding trained models we rely heavily on SHAP values and in particular on different types of plots that provide more context for interpretations. Therefore, we first explain how to interpret these plots, in particular Beeswarm plots (Section 4.3.1) and Waterfall plots (Section 4.3.2). Beeswarm plots are used to gain insights on a macro-level about how features impact predictions and how feature value ranges relate to class labels. In contrast, Waterfall plots focus on single documents and explain why it was predicted to have a specific class label.

4.3.1 Beeswarm Plots

According to our problem definition from Section 3.1.1, easier documents are represented with higher labels. According to Equation 2.1 this implies for the inter-

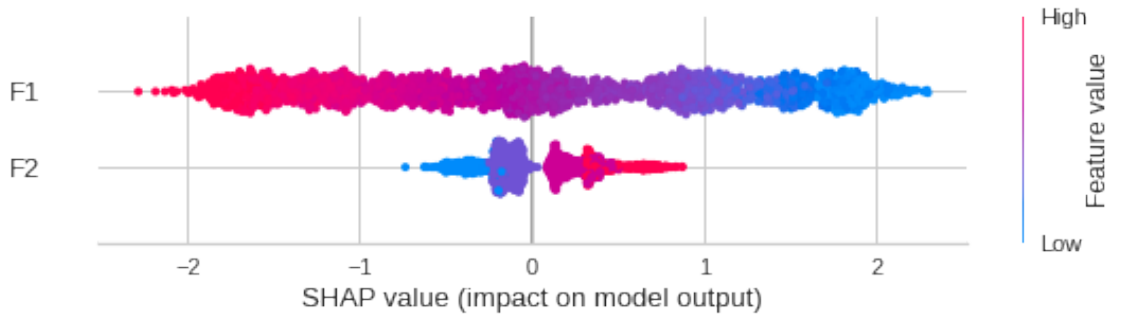


Figure 4.1: Sample Beeswarm plot. Feature F1 is the most important feature, while high values of F1 make the document harder (negative SHAP values), low F1 values shift the prediction for a document towards easier difficulty labels.

pretation of ϕ_i that a positive SHAP value for feature x_i shifts document x towards a higher label. In other words, a positive SHAP value for x_i indicates that this features makes x easier. Similarly, the larger $|\phi_i|$, the magnitude of SHAP value ϕ_i is, the more feature x_i contributes to the prediction of document x. Likewise, $|\phi_i| = 0$ indicates that feature x_i has no effect on the prediction of document x. Beeswarm plots reveal relationships between a feature and class labels, which correspond to difficulty levels in our problem setting. Similarly, they reveal how each feature affects predictions. One such Beeswarm plot is depicted in Fig. 4.1. Features are sorted with respect to importance in terms of the magnitude of the mean SHAP values from top to bottom. Thus, the most important feature is ranked at the top. For a feature all documents are plotted as points horizontally. If documents have the same feature value, they are stacked vertically, which may lead to bubbles characterizing distributions. This way the feature value distribution is visualized, while also revealing the relationship between the feature and the difficulty levels by coloring the feature values from high to low and indicating their effect towards making documents easier (positive SHAP values) or harder (negative SHAP values) on the x-axis. For example, feature F1 is the most important feature and high feature values occur with negative SHAP values, whereas positive SHAP values were computed for low feature values. This implies that high feature values of F1 make documents harder, whereas low feature values make documents easier.

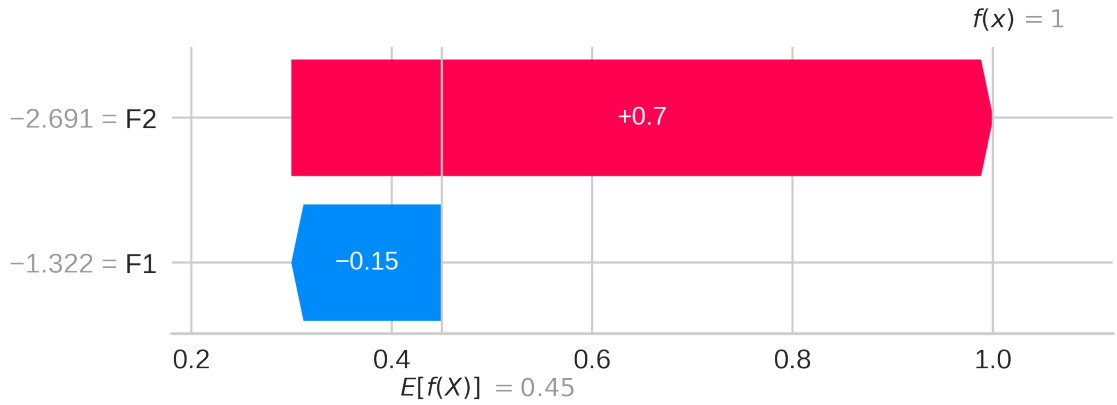


Figure 4.2: Sample Waterfall plot. Contribution of each feature (quantified by SHAP values) to the final prediction (label 1). The sum of all contributions plus the base value ($E[f(x)] = 0.45$) yields exactly the predicted label 1.

4.3.2 Waterfall Plots

Waterfall plots illustrate the impact, measured by SHAP values, of all features of a document on its predicted class. An example is given in Fig, 4.2. The grey values on the left side show the actual feature values of the document and features are sorted w.r.t. their impact on the document's predicted difficulty level. According to Eq. 2.1, the sum of all these SHAP values (plus the base value, which corresponds here to the mean class label) results exactly at this predicted difficulty level. For example, if the predicted label is 1 (=easy document), indicated by $f(x) = 1$ in the top right, summing up all SHAP values (ϕ_i in Eq. 2.1) plus the base value (ϕ_0 in Eq. 2.1), which is 0.45 in this case and is represented by $E[f(x)] = 0.45$ at the bottom of the plot, results in 1. Therefore, this Waterfall plot explains why a model predicted difficulty level 1 for this specific document.

4.4 Training Procedure

First, we allocate 80% of a dataset for training and tuning, and the remaining 20% for testing. Then training is done using 10-fold stratified cross-validation (CV), whereas the holdout set is only used for estimating model performance in an unbiased manner. Note that, in contrast to prior work [10, 11, 12], all difficulty levels of a document are part of the same fold or of the test set. This way information leakage is

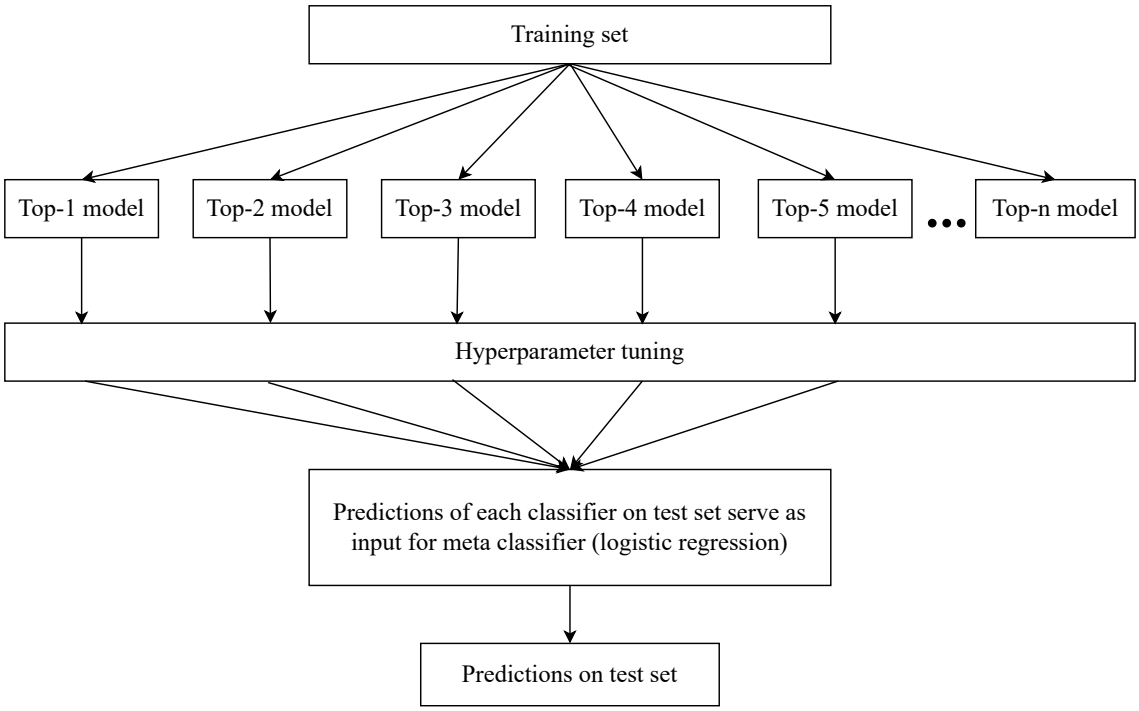


Figure 4.3: Procedure to build our ensemble model called ENSEMBLE from the top-5 models.

prevented. Otherwise different difficulty levels of the same document might exhibit similar features, which could overestimate the model performance in the worst case.

To achieve better performance, we also construct an ensemble as our main model to which we refer as ENSEMBLE. It is constructed according to the workflow described in Fig. 4.3. In total, we train 14 different models using all the 20 features described in Section 3.2.3 according to the training procedure described in this section. We then select the top-5 classifiers in terms of their F1-scores and tune their hyperparameters before using their predictions as features for the meta-classifier, which is logistic regression in our case. Note that with this workflow the top-5 models may change for ENSEMBLE depending on the dataset. The exhaustive list of all 14 classifiers is depicted in Table A.1.

4.5 Metrics

To evaluate model performance for predicting document difficulty, we report F1-score, precision, recall, and accuracy as all datasets are balanced. Since Newsela is

a multi-class dataset with $n = 5$ difficulty levels, we do not only want to quantify how accurately the model predicts the correct difficulty level, but also how well the model distinguishes easier from harder documents. Therefore, we compute pairwise accuracy between pairs of documents (d_j, d_k) with labels L_j and $L_k, L_j < L_k$, i.e. each pair contains first a harder document and then an easier one both describing the same news article. Specifically, for each news article with labels 0 (hard) - 4 (easy), we create 10 document pairs: $(L_0, L_1), (L_0, L_2), (L_0, L_3), (L_0, L_4), (L_1, L_2), (L_1, L_3), (L_1, L_4), (L_2, L_3), (L_2, L_4), (L_3, L_4)$ and compute their pairwise accuracy. Pairwise accuracy for a pair of documents (d_j, d_k) with predicted labels L_j, L_k is computed according to Equation 4.1 below:

$$\text{Pairwise accuracy}(L_j, L_k) = \begin{cases} 1, & \text{if } L_j < L_k \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

Note that we exploit in pairwise accuracy the knowledge that d_j is harder than d_k based on how we generated the pairs of documents (d_j, d_k). Therefore, pairwise accuracy measures how often d_j, the more difficult version of the same news article, is predicted to be harder than d_k, which is the easier version of the same news article. This metric considers a pair as correctly predicted, even though the predicted labels of d_j and d_k might differ from their true labels.

4.6 Baselines

In total, we employ four baselines. The first one is GCN [4]. Note that GCN produces six CEFR levels, whereas Newsela contains only five difficulty levels. Thus we map the six CEFR levels to the five difficulty levels, as follows: A1 $\rightarrow$ 4, A2 $\rightarrow$ 3, B1 or B2 $\rightarrow$ 2, C1 $\rightarrow$ 1, and C2 $\rightarrow$ 0. Mapping both B levels to difficulty level 2 is based on the fact that the difference in vocabulary size between both levels is minimal compared with the differences among other levels [57]. In Bio, we map $\{\text{B2}, \text{C1}, \text{C2}\} \rightarrow 0$ and $\{\text{A1}, \text{A2}, \text{B1}\} \rightarrow 1$. Another baseline we employ is built according to ENSEMBLE as described in Fig. 4.3, but utilizes only the top-1 model instead of the top-5 models. We refer to it as TOP-1. Since it is known that document length exhibits a high correlation with the difficulty levels on Newsela [10],

we also train a baseline exploiting only features related to document length. It also follows the same workflow described in Fig. 4.3, but uses only three features related to document length as input for all models. Those three features are the number of characters in a document, the number of words, and the number of sentences. We refer to this baseline as LEN. Last but not least, we also employ a baseline method called ALL, which is built according to Fig. 4.3 and utilizes all 20 features from Section 3.2.3 plus the three features from LEN.

We do not compare with either [10] or [12] because both studies use only 200 randomly selected articles from Newsela, which we cannot replicate. Moreover, in their experimental setup information could potentially leak into the test set because some versions of an article might be used for training, while others could be part of the test set. In the worst case, this experimental setup overestimates the performance of their proposed method.

4.7 Results

This section reports results for the different experiments. First, Section 4.7.1 reports the performance of ENSEMBLE in comparison to baseline methods to address RQ1. For RQ2 the importance of the different features of ENSEMBLE according to SHAP and permutation importance is evaluated in Section 4.7.2. Last but not least, we analyze relationships among features and how the features of ENSEMBLE interact using SHAP to address RQ3 in Section 4.7.3.

4.7.1 RQ1: Performance Comparison

Here we report the results obtained after training ENSEMBLE and all baselines according to Section 4.4. The results obtained on the test sets of Newsela and Bio are reported in Table 4.1. First of all, no overfitting was observed as the model performances on the validation (see Appendix A) and test set are similar. Secondly, the top-5 classifiers on Newsela are logistic regression, gradient boosting, linear discriminant analysis, random forest, and light gradient boosting machine. Likewise, top-5 classifiers on Bio are naive bayes, logistic regression, ridge, linear discriminant analysis, and random forest. Furthermore, ENSEMBLE outperforms the baseline

Table 4.1: Classifier performances in the multi-class setting.

Dataset	Model	F1-score	Precision	Recall	Accuracy	Pairwise Accuracy
Newsela	ENSEMBLE	0.4879	0.494	0.4865	0.4863	0.7865
	GCN	0.2612	0.4147	0.2439	0.2921	0.4706
	TOP-1	0.4676	0.4708	0.4676	0.4674	0.7918
	LEN	0.7698	0.7718	0.771	0.7708	0.8990
	ALL	0.7639	0.7655	0.7641	0.7639	0.9022
Bio	ENSEMBLE	0.6667	0.6604	0.6731	0.6667	0.3207
	GCN	0.4092	0.4510	0.3084	0.4626	0.0919
	TOP-1	0.6667	0.5	1.0	0.5048	0.3018
	LEN	0.4096	0.5667	0.3208	0.5377	0.1132
	ALL	0.6337	0.6531	0.6154	0.6476	0.3773

GCN on both datasets and performs at least on par with TOP-1 on both datasets. The main difference between Newsela and Bio is the performance of LEN and ALL. In Newsela both baselines exhibit superior performance and we observe performance increases up to 57% over ENSEMBLE in terms of F1-score. This result on Newsela is not unexpected given that sentences in Newsela were simplified one by one to obtain easier versions of a news article using the following set of valid operations for simplifying a sentence: deleting it, paraphrasing it, combining both operations, splitting the sentence up, or leaving it unaltered. Thus, operations either shorten the sentences or preserve their length. This inevitably introduces a bias towards shorter versions of news articles becoming easier. In contrast, ENSEMBLE slightly outperforms ALL and outperforms LEN by a large margin on Bio, which suggests that a length bias similar to Newsela does not exist. Pairwise accuracy of Newsela also indicates that ENSEMBLE manages to distinguish easier from harder news articles, but it fails to recognize the exact difficulty label, because its accuracy is substantially lower than the respective pairwise accuracy.

We further investigate how ENSEMBLE performs for different pairs of difficulty levels. First, we analyze accuracy and pairwise accuracy in Fig. 4.4 for Newsela only because this analysis is redundant for Bio. Then we continue examining the

remaining metrics F1-score, precision, and recall for Newsela (Fig. 4.5) and Bio (Fig. 4.6) for each difficulty level separately.

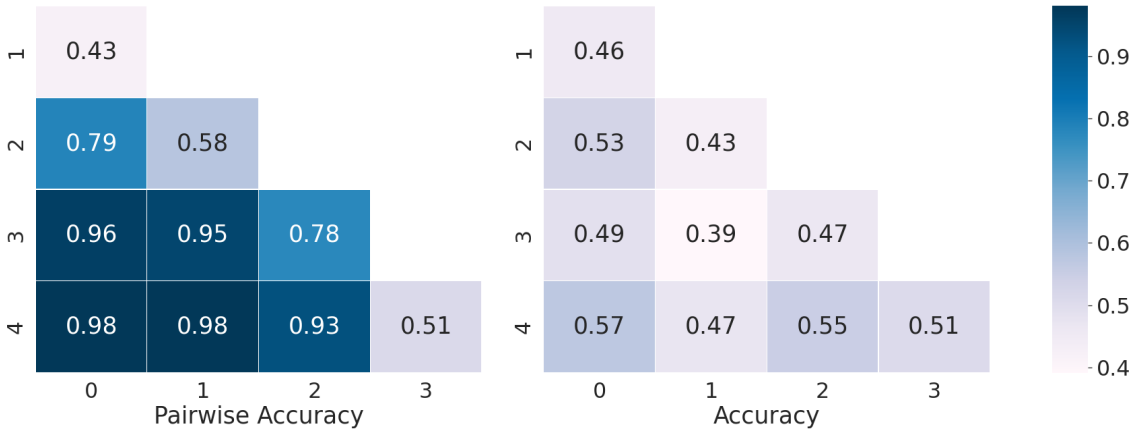


Figure 4.4: Performance of ENSEMBLE on Newsela comparing pairwise accuracy and accuracy.

Pairwise accuracy and accuracy for ENSEMBLE are shown for each pair of difficulty levels in Fig. 4.4. Most importantly, the performance of ENSEMBLE in terms of pairwise accuracy improves the further the difficulty levels in a pair of documents are apart, which suggests that ENSEMBLE does separate easier from harder documents, although the accuracies for all classwise comparisons look similar, with only pairs involving either difficulty levels 1 and 3 or levels 1 and 2 performing noticeably worse. ENSEMBLE achieves a pairwise accuracy of at least 95% for pairs with one easier document (either difficulty level 3 or 4) and a harder document (either difficulty level 0 or 1), while such pairs contain no documents with adjacent difficulty levels like 0 and 1, 1 and 2, etc. That also explains why pairwise accuracy involving difficulty level 2 tends to be lower - only a combination of levels 2 and 4 results in accurate predictions because both levels are not adjacent. For pairs with non-adjacent difficulty levels the combination involving 0 and 2 yields the lowest pairwise accuracy with 79%, whereas the lowest pairwise accuracy is 43% for pairs with documents from the adjacent combination of difficulty levels 0 and 1. Moreover, it seems that distinguishing pairs with easier difficulty levels (which include levels 2, 3 and 4) is easier than separating harder pairs (which include levels 0, 1 and 2). This suggests that more features discriminating difficult documents could enhance the performance of ENSEMBLE.

Focusing on the classwise performances of Newsela in Fig. 4.5 and of Bio in Fig. 4.6,

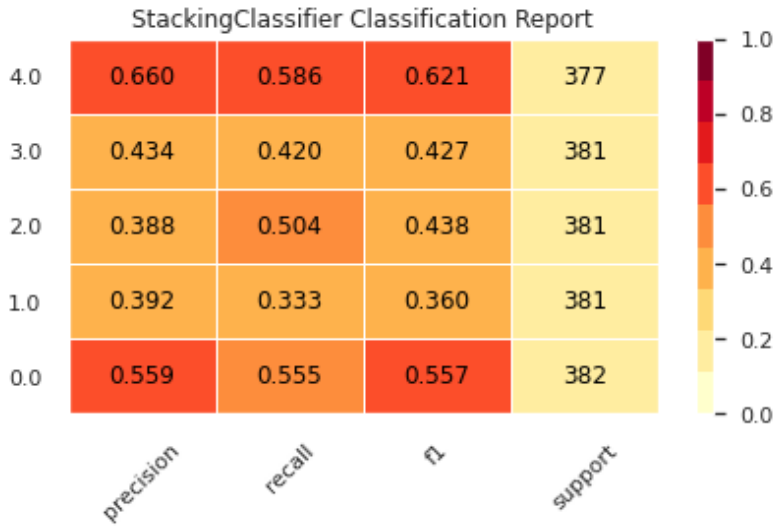


Figure 4.5: Classwise performance of ENSEMBLE on Newsela.

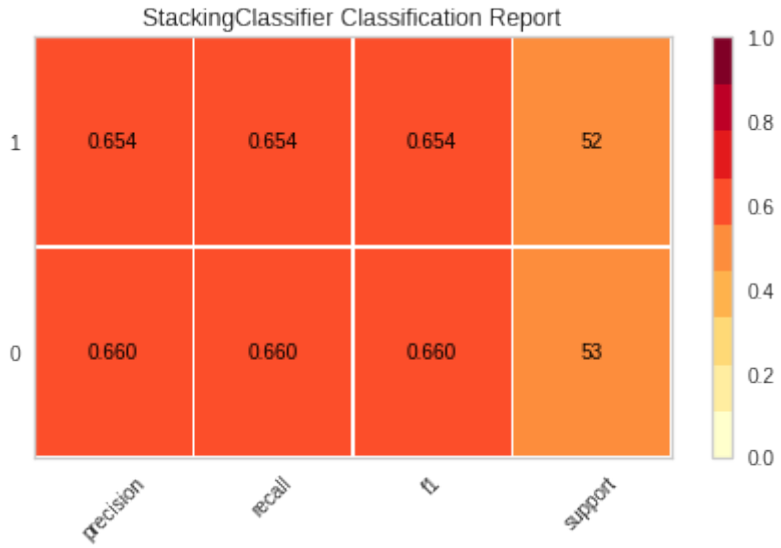


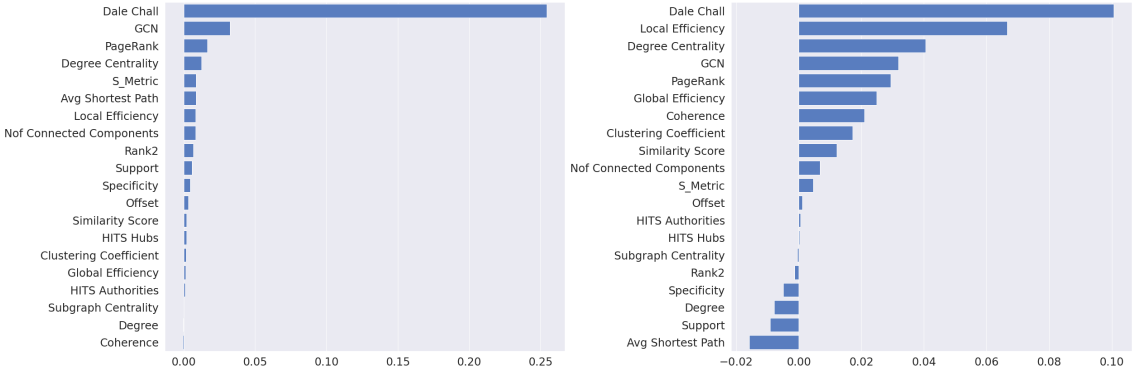
Figure 4.6: Classwise performance of ENSEMBLE on Bio.

we find that ENSEMBLE exhibits the same performance on Bio for both difficulty levels. In contrast, on Newsela performances vary depending on the difficulty level. ENSEMBLE predicts the easiest (level 4) and hardest (level 0) difficulty levels more reliably, whereas the distinction of the remaining levels seems more challenging. This suggests that our extracted features capture differences between the easiest and hardest difficulty levels best, but are insufficient to separate difficulty levels 1-3 correctly.

4.7.2 RQ2: Most Important Features for Document Difficulty

In this section we estimate feature importance in ENSEMBLE with two methods. First, we report permutation importances. Then we report importance rankings obtained with SHAP.

Permutation Importance



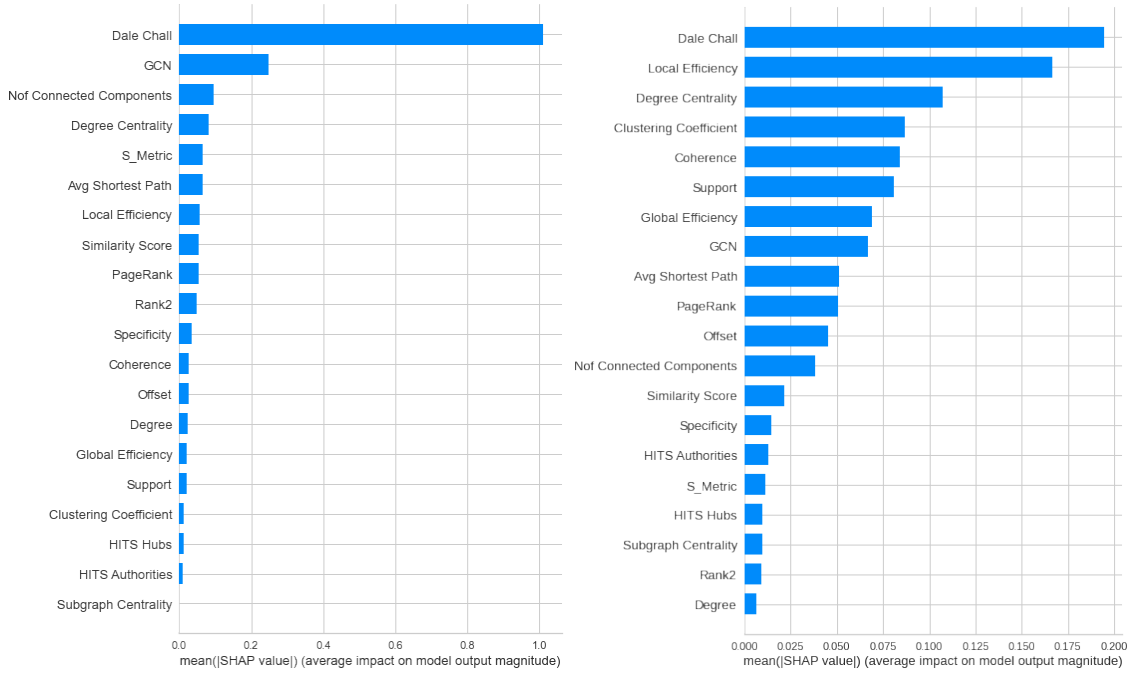
(a) Feature Importances on Newsela. (b) Feature Importances on Bio.

Figure 4.7: Feature importances according to permutation importance per dataset.

The results of estimating feature importance according to permutation importance are displayed in Fig. 4.7. Most notably, the importance of features varies substantially across datasets. The top-5 features are Dale-Chall, GCN, Degree Centrality, PageRank, and Local Efficiency when sorting feature importances according to average ranks. Similarly, on average the top-5 least important features are Degree, Subgraph Centrality, HITS Authorities, Support, HITS Hubs and Specificity are tied. The features whose importance varies the most across both datasets are Avg. Shortest Path, Coherence, and Global Efficiency. Moreover, on Bio six features are deemed unimportant by permutation importance, while this is only the case for three features on Newsela. Interestingly Dale-Chall is considered to be the most important feature on both datasets by a large margin, which suggests that language difficulty overlaps heavily with conceptual text complexity. This is corroborated by the fact that GCN, which also captures language difficulty, is consistently among the four most important features. In contrast, inherent concept difficulty, encoded

by features extracted from the graph structure of DBpedia, seems to have a higher impact on predicting conceptual text complexity on Bio as they exhibit higher importance scores on Bio. One possible explanation for this observation could be the scope of the datasets: while Newsela is comprised of news articles about the state of the world, Bio is tailored towards teaching specific concepts. This way it might be easier for DBpedia to identify and link these concepts to DBpedia entries.

SHAP

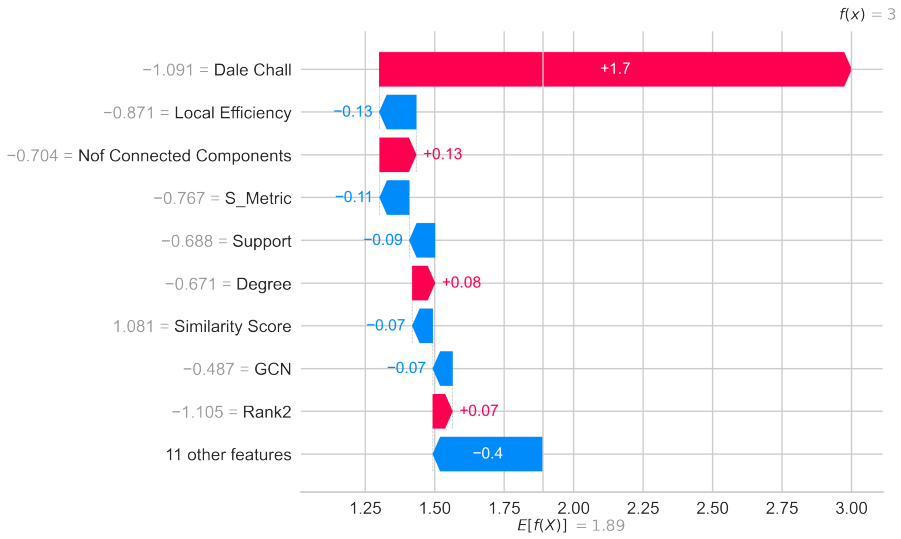


(a) Newsela. (b) Bio.

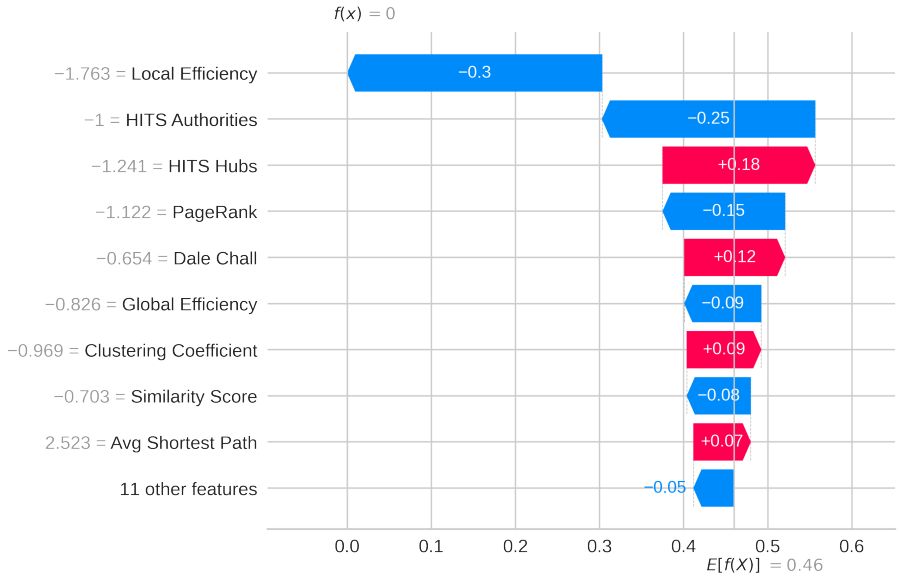
Figure 4.8: Feature importances according to SHAP per dataset.

Fig. 4.8 depicts the importance of each feature in ENSEMBLE on Newsela and Bio according to SHAP. It is immediately visible that Dale-Chall dominates the ranking of feature importances on Newsela. Other features contribute much less to correct predictions. This observation suggests that most predictions on Newsela depend mainly on Dale-Chall - if this feature does not capture the actual difficulty level of a news article well, a prediction will likely be incorrect. In contrast, a larger subset of important features exists on Bio which is utilized for classification, and which yields more robust results. This interpretation is also consistent with the results from Section 4.7.1, where ENSEMBLE achieved better performances on Bio compared

with Newsela.



(a) Misclassified Newsela article (true difficulty level=0, predicted level=3). Dale-Chall is largely responsible for the misclassification.



(b) Many features contribute to the correct prediction (difficulty level=0) of the Bio document, despite Dale-Chall shifting the prediction towards the opposite difficulty level.

Figure 4.9: Impact of Dale-Chall on predicting document difficulty for specific documents.

We also selected one document from Newsela and one from Bio to highlight this problem with Dale-Chall. The Waterfall plots are shown in Fig. 4.9. The true label of the Newsela document is 0 (most difficult) and it is misclassified as 3. Dale-Chall

has the largest (false) contribution to the prediction towards the false direction. Therefore, the misclassification can be largely attributed to Dale-Chall. In contrast, in Bio multiple features have higher importance and although Dale-Chall shifts the prediction in the wrong direction, the combination of the other features still leads to the correctly predicted difficulty level, which is 0 here.

Comparing the consistency of the rankings of the SHAP values in Fig. 4.8 on both datasets shows that Dale-Chall, Degree Centrality, Local Efficiency, and GCN are the most important features when sorting them according to their average ranks. The least important ones are Subgraph Centrality, HITS Hubs, Degree, HITS Authorities, and Rank2. The features whose importance varies the most on both datasets are Clustering Coefficient, S-Metric, and Support.

Consistency between SHAP and Permutation Importance

When considering the average ranks of all features in terms of their importance according to SHAP and permutation importance over both datasets, both methods agree on the impact of some features on predicting document difficulty. Most importantly, both regard Dale-Chall as the most important feature on each dataset. The other important set of features that both methods agree on comprises GCN and Degree Centrality. Similarly, the trend that more features matter on Bio than on Newsela is consistent in both methods. In terms of the least important set of features both methods agree on Subgraph Centrality, Degree, HITS Hubs, and HITS Authorities. The five most important features according to SHAP and permutation importance, when considering their average ranks, are Dale-Chall, Degree Centrality, GCN, Local Efficiency, and PageRank. Similarly, the five least important ones are Subgraph Centrality, Degree, HITS Authorities, HITS Hubs, and Rank2. The features with the highest variance in terms of rankings are Coherence, Clustering Coefficient, Avg. Shortest Path, Support, and S-Metric. Thus, it is unclear how important these features, all related to the factor inherent concept difficulty, are for predicting document difficulty, while both features quantifying language difficulty, namely GCN and Dale-Chall, are considered important.

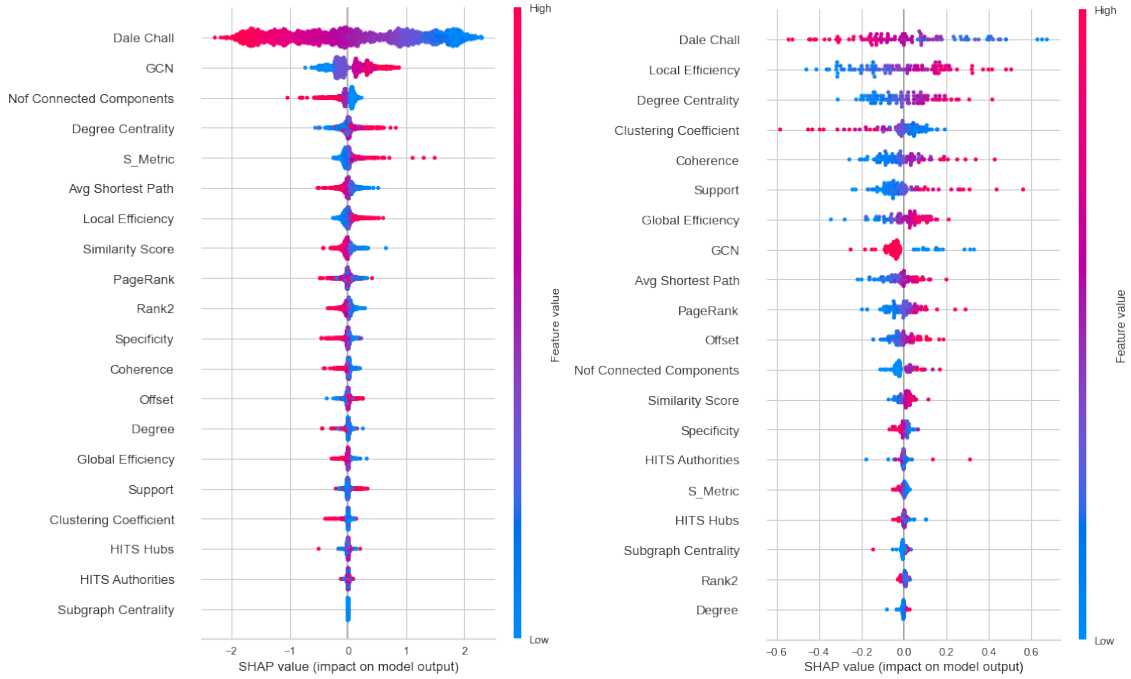
4.7.3 RQ3: Relationships Among Features

To better understand how features of easier and harder documents differ in ENSEMBLE, we first analyze relationships between features and difficulty level using Beeswarm plots. Since features may also interact with one another, we examine feature interactions separately afterwards. Last but not least, we investigate if relationships between features and difficulty levels differ between correctly and incorrectly classified documents. In this section we limit the analysis to Newsela, because SHAP results for Bio turned out to be unstable when re-running experiments with different splits of Bio, which is demonstrated in Appendix B. This instability results from the small dataset size. However, the performance of Bio remains stable across different splits of the dataset.

Relationships between Features and Difficulty Level

SHAP also gives insights about the relationships between features and class labels with a Beeswarm plot. First, we examine the top-6 features in terms of their relationships with difficulty levels on both datasets as these features will be used for further analyses later. Note that we only include Bio for reference, because SHAP results are unstable as explained above.

In Fig. 4.10a the effect of how specific ranges of feature values affect predicted difficulty levels. Higher values of Dale-Chall make documents harder, whereas lower values make them easier, which is expected based on how Dale-Chall values are defined. The Dale-Chall values of documents are also relatively evenly distributed and more extreme values have a higher impact on predictions, while medium values barely influence the difficulty level. High values of GCN make a document easier, while lower values make a document harder, which is expected because harder CEFR levels are expressed with lower values, whereas easier CEFR levels are encoded as higher values. For most documents GCN exerts a low impact on predictions because most documents exhibit low SHAP values. In terms of Nof Connected Components, higher values make documents harder and lower values simplify them. However, most of the time this feature has little to no impact on the predicted difficulty level. If it has influence, mainly high values affect predictions. A higher Nof Connected



(a) Newsela.

(b) Bio.

Figure 4.10: Influence of feature values on prediction according to SHAP per dataset. Positive SHAP values for a feature indicate that this feature shifts predictions towards easier difficulty levels, whereas negative SHAP values for a feature indicate that this feature shifts predictions towards harder difficulty levels. In short, positive SHAP values indicate that a feature makes a document easier, while negative SHAP values show that a feature makes a document harder.

Components value in document d_i indicates that more diverse concepts are discussed which are not linked in the constructed undirected graph G_i of d_i (see Section 3.2.2) via a dbo:wikiPageWikiLink edge. Hence, covering more diverse concepts in the same document makes this document harder. Low Degree Centrality values make documents harder, whereas high values simplify documents. But in most documents this feature has only marginal influence on predictions, and only in case of higher values it exerts a larger influence in making documents easier. S-Metric exhibits the same pattern as Degree Centrality. High Avg. Shortest Path values indicate harder documents, while lower values shift predictions towards easier difficulty levels. For most documents this feature does not affect classifications.

In Fig. 4.10b Dale-Chall follows the same pattern as in Newsela. Low values in Local

Efficiency make documents harder, whereas higher values simplify it. Degree Centrality exhibits the same pattern as in Newsela. Higher Clustering Coefficient values make documents harder and lower ones simplify documents. Higher values affect classification more profoundly. Lower Coherence values make documents harder and higher ones make them easier. Support follows the same pattern as Coherence.

With Fig. 4.10a we can analyze the hypotheses we formulated for every feature in Section 3.2.3. We focus only on Newsela since it is bigger and therefore results are more robust. As shown in 4.10a, our hypotheses about Dale-Chall, GCN, Nof Connected Components, Degree Centrality, S-Metric, Avg Shortest Path, Local Efficiency, Specificity, Offset, Support, Clustering Coefficient are correct. Since HITS Authorities, HITS Hubs and Subgraph Centrality have almost no impact on predictions, we cannot verify the correctness of our hypotheses. In contrast, our hypotheses about Similarity, PageRank, Rank2, Coherence, Degree, Global Efficiency seem incorrect. Since Similarity and Rank2 are related to the confidence that keywords from the raw text are linked to corresponding DBpedia entries, our hypotheses about those features could be wrong due to the fact that this disambiguation about linking to DBpedia may not be an important factor for predicting difficulty levels. The hypotheses about PageRank and Degree could be incorrect since both are related to the number of edges of concepts in documents and it might not affect the difficulty level contrary to our initial thought. Difficult documents could also contain popular concepts that are difficult themselves. Coherence indicates the average number of mutual categories. Since some concepts could share many common categories while others share few, averaging them could lead to artificially low Coherence scores. Considering only the most related concepts when computing Coherence might be a more reliable approach. Since the computation of Global Efficiency takes the distant parts of the graph into account, it is more sensitive to outliers than Local Efficiency. Thus, some distant part of the graph could artificially lower Global Efficiency. Therefore, Local Efficiency is more reliable than Global Efficiency in terms of prediction contribution.

Interactions Among Features

We focus only on the interactions of the top-6 features according to Section 4.7.2 because they have the biggest impact on predictions. For each of these top-6 feature we consider the feature with the strongest interaction and create a dependence plot. In these plots an interaction between features A and B means that effects of feature A depend on effects of feature B.

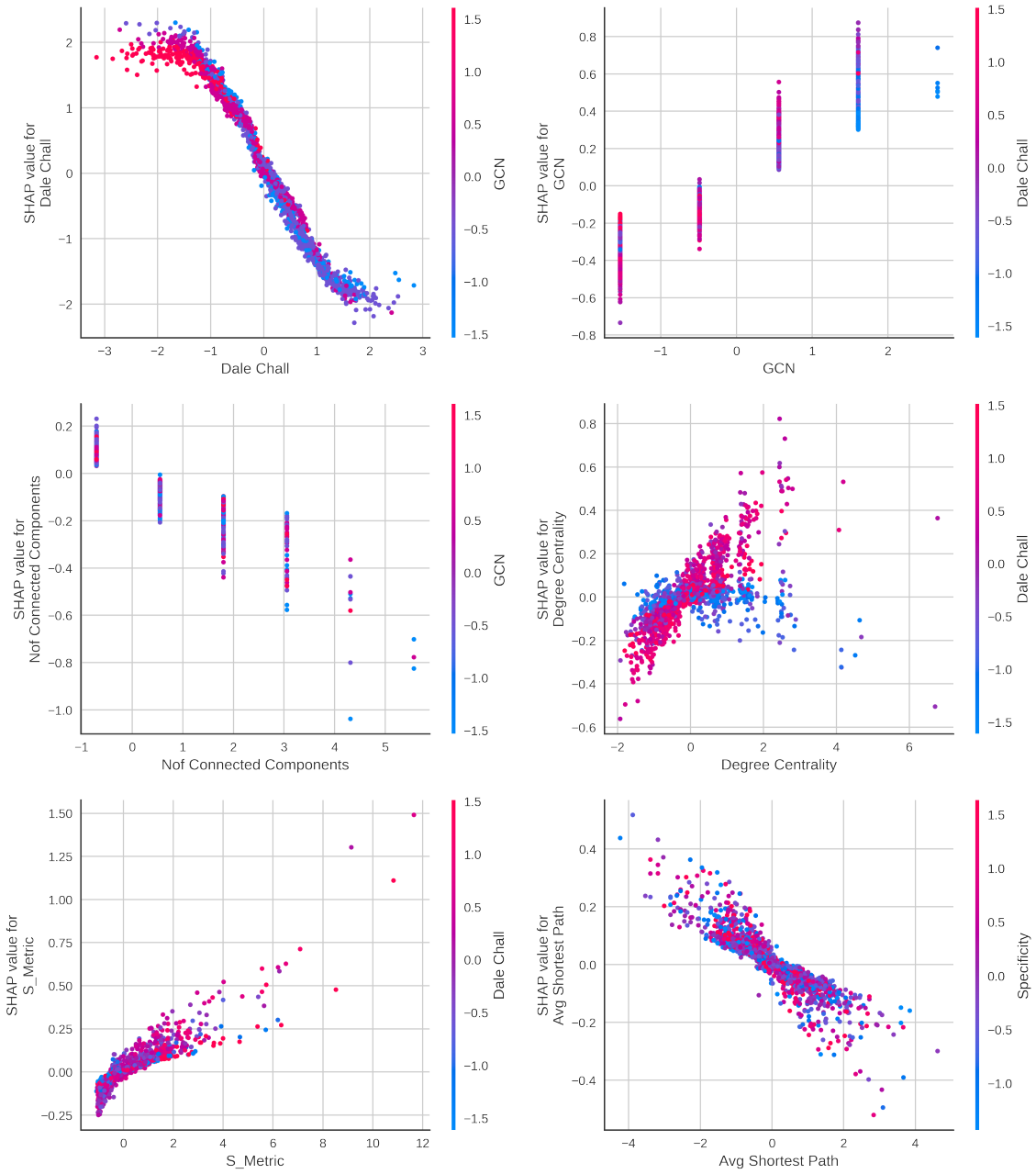


Figure 4.11: Interactions of the top-6 features in Newsela.

The results for Newsela are shown in Fig. 4.11. We observe that Dale-Chall interacts

with GCN. Since for a fixed x-value, which corresponds to a normalized Dale-Chall value, its y-value, i.e. the impact of the Dale-Chall value on prediction, exhibits a spread, suggesting that at least one other feature interacts with Dale-Chall. In this specific case, GCN explains most of the variance in spread of the Dale-Chall values. Specifically, for low Dale-Chall values the GCN values tend to be higher. When combining that information with the SHAP values on the y-axis, we can conclude that low Dale-Chall values that have high GCN values make a document easier. This observation is consistent for both features in isolation: low Dale-Chall values simplify a document and high GCN values make a document easier as well. Both features combined distinguish easier documents more accurately indicating that GCN and Dale-Chall complement each other and capture different aspects of language difficulty, especially for easier documents. This finding is also in line with Fig. 4.4, where ENSEMBLE manages to separate easier documents more reliably than hard ones. While Dale-Chall considers only the percentage of difficult words and average number of words per sentence, GCN exploits word frequency and difficulty among other features with a high-dimensional representation per word, which seems to encode different information than Dale-Chall.

GCN interacts with Dale-Chall, which we already discussed above.

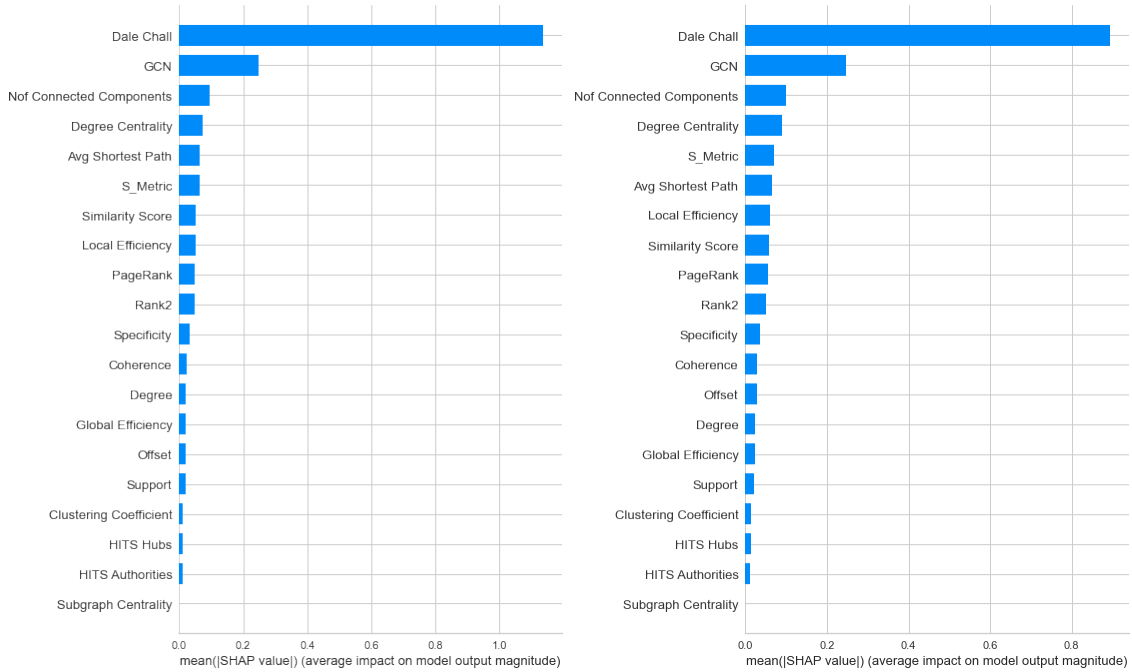
When examining the interactions between Nof Connected Components and GCN, we do not observe a easily interpretable interaction which is indicated by the almost equally distributed feature values of GCN within each value of Nof Connected Components. It may be due to both features being discrete that the interaction is hidden.

Degree Centrality interacts with Dale-Chall. Higher Degree Centrality values make a document easier if Dale-Chall values are high, i.e. well-connected (=popular) concepts make a document easier if the text in the document is difficult to read. However, once Dale-Chall values are low, which means a document is easily readable, the popularity of a concept does not provide any additional information for the document difficulty as SHAP values of Degree Centrality are close to 0. This suggests that readability dominates concept popularity if one has to make a trade-off between using easy or hard language versus popular or unpopular concepts and that it is more

important to write the text in an easy language.

When examining the strongest interaction between S-Metric versus Dale-Chall and Avg. Shortest Path versus Specificity, it seems that there is no clear interaction pattern.

Differences between Correctly and Incorrectly Classified Documents



(a) Feature importances of correctly classified articles.

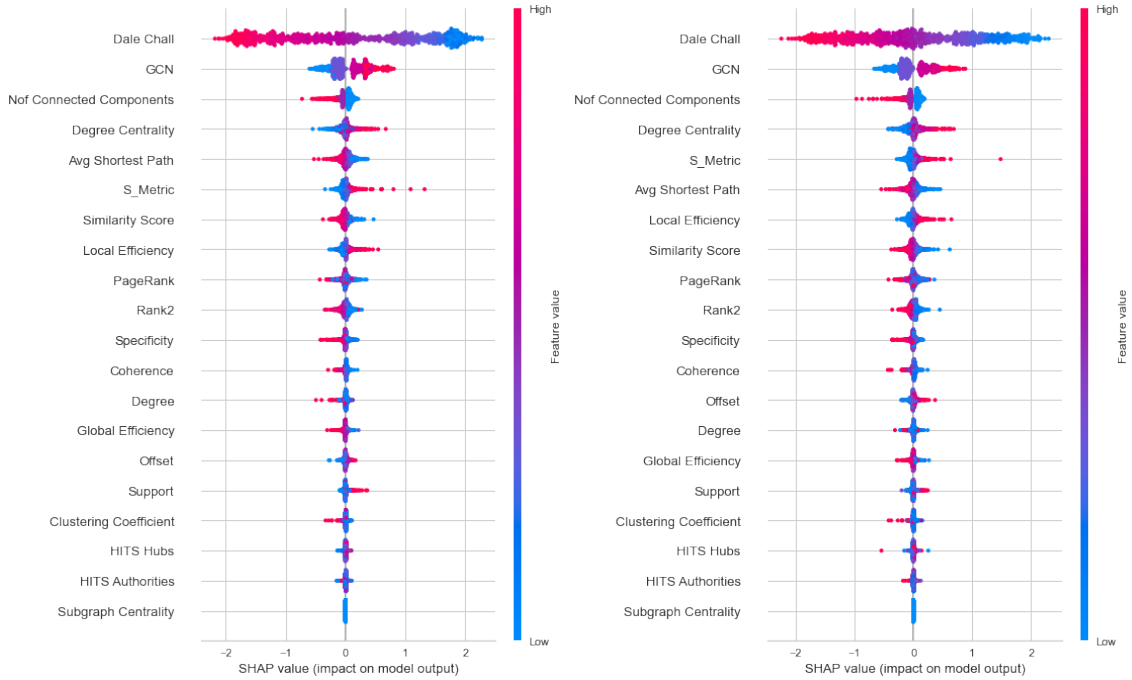
(b) Feature importances of misclassified articles.

Figure 4.12: Global feature importances according to SHAP based only on correctly/incorrectly classified Newsela articles.

To better understand why some documents are classified incorrectly, we compute SHAP values separately for correctly and incorrectly predicted documents. This way all features can be ranked according to their importances using only correctly classified documents on the one hand and using only incorrectly classified documents on the other hand. Potential differences could explain the misclassifications. Similarly, two separate Beeswarm plots can be generated based on the computed SHAP values to identify potential differences in terms of relationships between features and difficulty levels.

First, we focus on the global feature importances of Newsela according to SHAP

values separately for correctly and incorrectly classified documents. As shown in Fig. 4.12a Dale-Chall has a lower importance when considering SHAP of misclassified articles only compared to SHAP of correctly classified articles in Fig. 4.12b.



(a) Correctly classified articles.

(b) Incorrectly classified articles.

Figure 4.13: Beeswarm plots using only correct or incorrect classifications in Newsela.

In Fig. 4.12 we display Beeswarm plots of correctly and incorrectly classified documents. Overall, the relationships between features and difficulty level are identical on both datasets. Given that most features have little to no impact on predictions, which is indicated by the large number of documents around SHAP value 0, we focus on differences in Dale-Chall as it is the most important feature by a large margin. As can be seen in Fig. 4.12b, many missclassified documents have medium Dale-Chall values whose contribution to classification is negligible due to SHAP values being close to 0. In contrast, most correctly classified documents exhibit extreme Dale-Chall values, which shift predictions either towards easier or harder predictions, which is depicted in Fig. 4.12a. This suggests that most of the correctly classified documents tend to be the easiest and most difficult documents where Dale-Chall suffices to capture the correct difficulty level. Applying the same argument to the incorrectly classified documents implies that these documents contain more

likely medium Dale-Chall values, which is not informative enough for steering the predicted class label in the correct direction. Other features should help in that regard, but fail, which hints at a lack of features that reliably discriminate documents with medium Dale-Chall values, which is in line with Section 4.7.1, because ENSEMBLE has problems distinguishing documents with intermediate difficulty levels 1-3. Moreover, for the other features other than Dale-Chall we would expect to observe more differences between correctly and incorrectly classified documents if they impacted predictions. Overall, a possible explanation for the minor differences observed between correctly and incorrectly classified documents using SHAP values could be that the majority of our features are insufficient to distinguish the difficulty levels of news articles.

4.8 Discussion

Despite having conducted experiments on Newsela and Bio for research questions RQ1-RQ3, we report results for RQ2 and RQ3 only on Newsela, because the results on Bio are unstable given its small size. Although we report results on Bio for RQ2, we do not compare them with Newsela to avoid drawing any incorrect conclusions and treat the results on Bio as preliminary results. However, for RQ1 performances measured are robust on Newsela and Bio in that splitting the dataset randomly into training and test set does not affect performances.

Most importantly, our experiments reveal that the factor language difficulty is more important than inherent concept difficulty for predicting document difficulty, because Dale-Chall is ranked consistently first in terms of feature importance independent of the dataset and method for estimating feature importance. This seems consistent with the intuition that one can only learn what one reads. This finding holds on Newsela and also on Bio according to Section 4.7.2, even when splitting Bio differently (see Appendix B). In line with that, it seems more important to use easier language than well-known concepts when writing a document as pointed out for Newsela in Section 4.7.3. It is unclear if this finding is reliable or not due to the way Newsela was created. As explained in Section 4.1, Newsela was created for sentence simplification, which introduces an artificial length bias s.t. easier articles

become shorter, because human experts removed or paraphrased more difficult sentences to simplify them in order to ensure that sentences stay aligned across different difficulty levels. Moreover, simplified versions obey the Lexile readability formula, which incentivized human experts to use words that are considered easier according to that specific formula. Thus, we speculate that this language bias could lead to overestimated feature importances for features related to language difficulty, which are GCN and Dale-Chall, on Newsela as reported in Section 4.7.2. If this were the case, language difficulty would still play an important role in predicting document difficulty, but the importance of inherent concept difficulty could increase.

It is known that Newsela has a length bias, e.g. [10] explicitly note that no features without normalization may be used as it would leak information about the respective document difficulty otherwise. In our experiments we also observe this problem in the baselines LEN and ALL, exploiting document length, as shown in Table 4.1, because these baselines outperform ENSEMBLE by a large margin. In contrast, this behavior does not occur in Bio, where ENSEMBLE slightly outperforms the aforementioned baselines. This is most likely due to collecting explanations for the same biological concept from two independent resources, but it suggests that this type of dataset is more suitable as a benchmark dataset for predicting document difficulty. Unfortunately, Bio is too small for quantifying feature importance reliably as shown in Section 4.7.3, but it shows that alternative datasets for estimating document difficulty can be constructed.

Based on our experimental results obtained in Newsela, it seems also worth exploring additional features in the future. While we employ 18 features related to inherent concept difficulty, we utilize only two features related to language difficulty. Yet, language difficulty turns out to be more impactful than inherent concept difficulty, but this finding might be affected by the implicit language bias of Newsela. Nevertheless, GCN and Dale-Chall seem to complement each other according to Section 4.7.3, thus including additional features related to language difficulty could be promising. In terms of inherent concept difficulty, Degree Centrality, Local Efficiency, and PageRank perform well according to permutation importance and SHAP. The features that seem the least promising are Subgraph Centrality, Degree, HITS

Authorities, HITS Hubs, and Rank2. While the low importance of Rank2, which is related to matching extracted keywords to DBpedia entries, is conceivable, the other four features are more surprising. While PageRank is important, both features related to HITS are not, and similarly Degree shares similarities with Degree Centrality, yet it barely affects document difficulty. We offer three possible explanations for these results.

The first explanation relates to our trained model called ENSEMBLE, because we analyze feature importances w.r.t. ENSEMBLE. While it distinguishes easier from more difficult documents according to pairwise accuracy in Fig. 4.4, it fails to detect subtler differences between difficulty levels 1-3 as shown in Fig. 4.5. These problems also surface when estimating feature importances using SHAP and permutation importance, as both evaluate importance w.r.t. the trained model, ENSEMBLE in our case. Hence, we draw similar conclusions in Section 4.7.1 and in Section 4.7.3 where we find that ENSEMBLE separates easier documents more easily than difficult ones.

The second possible explanation relates to the choice of datasets. While Bio is of preliminary nature, Newsela might not be an ideal benchmark dataset for predicting document difficulty due to the language and length biases.

The third explanation relates to the scope of Bio and Newsela. Bio is geared towards teaching specific biological concepts. Hence, it is structured in a way that related concepts are taught together, ideally one at a time to reduce the mental load of learners. Our features might reflect this difference in scope of both datasets. On the one hand, Newsela news articles are all grounded in the Lexile readability measures [55], i.e. articles have an appropriate Lexile score for the respective target audience. This also explains why GCN and Dale-Chall are the two most important features on this dataset according to Section 4.7.2. On the other hand, background knowledge becomes more important in Bio, as observed in Section 4.7.2, where features related to inherent concept difficulty tend to contribute more, so that correct predictions rely less on Dale-Chall alone, which makes the predictions more robust and is also reflected in overall better performances of ENSEMBLE in Bio as we showed in Section 4.7.1.

Chapter 5

Conclusion and Future Work

In this thesis we set out to examine how different features affect document difficulty. After modeling document difficulty to be comprised of two factors, namely linguistic difficulty and inherent concept difficulty, we utilized an DBpedia as an external knowledge base to extract subgraphs that were utilized for estimating inherent concept difficulty. Similarly, we extracted features gauging linguistic difficulty in terms of lexical and syntactical text properties. This resulted in 20 features utilized for training a supervised ensemble classifier on two datasets to obtain more robust results. Those datasets comprise the popular Newsela dataset as well as a newly generated one about biology for high school students and biology major students. With the help of the trained ensemble classifier we investigated the importance of features and interplay thereof using SHAP. It turned out that linguistic difficulty is more important than inherent concept difficulty for predicting document difficulty. Due to Newsela being originally devised for sentence simplification, our new dataset is more appropriate for estimating document difficulty as the easier and harder version of documents were written independently, whereas in Newsela sentences were gradually simplified by paraphrasing and removing redundant information to keep sentences aligned across multiple difficulty levels, which introduces an artificial bias towards shorter sentences and documents being easier. We demonstrated this bias experimentally, because our supervised ensemble model trained only on document length outperformed our ensemble method by a large margin. Adding these features to the ensemble method enhanced its performance to a similar degree. In contrast, the effect of document length in the new dataset is smaller as the model trained on

features related to document length did not outperform our ensemble. This result is consistent with intuition, because longer documents may also be easy, e.g. adding more examples for clarification reduces the difficulty of a document. Hence, Newsela is not ideal for estimating document difficulty due to the artificial bias in terms of document length. Another bias in Newsela is that the articles comply with a specific readability measure, which potentially inflates the importance of features related to readability.

This observation raises the question how generalizable our results are for three reasons. First, results obtained with SHAP are model-dependent, i.e. our reported results hold only for our specific model and could get affected by replacing either the training procedure, the classifier, or both. Secondly, the results from Newsela might not generalize due to the artificial bias. Although most of the results were also observed in our biology dataset, this could be a coincidence due to the small size of the dataset. Hence, the results on the biology dataset must be regarded as preliminary.

This shortcoming directly motivates our plan to create more extensive datasets for the educational domain. While we have started this endeavor with the biology dataset, additional domains must be covered as well, most notably chemistry and computer science. One potential problem with other STEM fields like physics or mathematics is that these rely heavily on equations. Thus mapping them to DBpedia might prove to be challenging. Similarly, capturing features related to linguistic difficulty would be impacted by the sheer amount of equations instead of textual descriptions. Another avenue for future research is integrating our method into an e-learning platform like [1] to recommend more learning materials of a difficulty level that is appropriate for individual learners. Estimating document difficulty could also have benefits for prerequisite detection, in which the goal is to predict all prerequisites for a given target concept. We speculate that document difficulty could be another feature for determining prerequisites, because it is conceivable that prerequisites are easier than a target concept. Hence, investigating this relationship could prove to be promising.

In this thesis we assumed document difficulty to be of general nature, i.e. everyone

perceives a document to be equally difficult. However, in practice this is not the case and document difficulty is subjective [58]. Therefore, one would have to incorporate a user model that encodes their background knowledge and abilities to estimate document difficulty per individual. Another way we can imagine to mitigate subjective document difficulty, at least to some extent, is by ensuring that individuals have the same prior knowledge before reading a specific document. This can be accomplished by explicitly providing prerequisites for each document. But this solution seems mainly applicable in an educational environment such as e-learning platforms where specific knowledge is taught in the form of courses.

Appendix A

Analysis of Overfitting

Table A.1: Performances of all 14 classifiers using default hyperparameters on the validation set of Newsela. Majority always predicts the majority label as a baseline.

	Model	F1-score	Precision	Recall	Accuracy
0	Logistic Regression	0.4709	0.4712	0.4768	0.4762
1	Gradient Boosting	0.4696	0.4712	0.4720	0.4715
2	Linear Discriminant Analysis	0.4692	0.4698	0.4744	0.4738
3	Random Forest	0.4558	0.4559	0.4601	0.4595
4	Light Gradient Boosting Machine	0.4549	0.4559	0.4572	0.4566
5	Extra Trees	0.4423	0.4410	0.4494	0.4488
6	Ada Boost	0.4364	0.4353	0.4477	0.4469
7	Decision Tree	0.3778	0.3798	0.3780	0.3776
8	SVM - Linear Kernel	0.3736	0.3744	0.4119	0.4107
9	K Neighbors	0.3712	0.3764	0.3735	0.3733
10	Naive Bayes	0.2788	0.4854	0.3406	0.3413
11	Ridge	0.2754	0.3453	0.3990	0.3973
12	Quadratic Discriminant Analysis	0.2615	0.4460	0.3297	0.3305
13	Majority	0.0671	0.0403	0.2000	0.2006

Table A.2: Performances of the top-5 classifiers after tuning on the validation set of Newsela.

	Model	F1-score	Precision	Recall	Accuracy
0	Logistic Regression	0.4715	0.4715	0.4777	0.4771
1	Gradient Boosting	0.4704	0.4702	0.4748	0.4742
2	Linear Discriminant Analysis	0.4695	0.4702	0.4748	0.4742
3	Random Forest	0.4620	0.4621	0.4680	0.4674
4	Light Gradient Boosting Machine	0.4686	0.4700	0.4722	0.4716

Table A.3: Performances of the top-5 classifiers after tuning on the test set of Newsela.

	Model	F1-score	Precision	Recall	Accuracy
0	Logistic Regression	0.4676	0.4708	0.4676	0.4674
1	Gradient Boosting	0.4651	0.4658	0.4656	0.4653
2	Linear Discriminant Analysis	0.4729	0.4766	0.4733	0.4732
3	Random Forest	0.4743	0.4748	0.4766	0.4763
4	Light Gradient Boosting Machine	0.4613	0.4628	0.4624	0.4621

Table A.4: Performances of all 14 classifiers using default hyperparameters on the validation set of Bio. Majority always predicts the majority label as a baseline.

	Model	F1-score	Precision	Recall	Accuracy
0	Naive Bayes	0.6010	0.4935	0.8756	0.4855
1	Logistic Regression	0.5819	0.6016	0.5821	0.5837
2	Ridge	0.5793	0.6054	0.5750	0.5840
3	Linear Discriminant Analysis	0.5740	0.5939	0.5750	0.5757
4	Random Forest	0.5694	0.5892	0.5654	0.5805
5	K Neighbors	0.5612	0.5810	0.5564	0.5678
6	Ada Boost	0.5401	0.5396	0.5583	0.5395
7	Extra Trees	0.5357	0.5600	0.5333	0.5475
8	SVM - Linear Kernel	0.5327	0.5206	0.5571	0.5057
9	Majority	0.5315	0.3980	0.8000	0.4940
10	Light Gradient Boosting Machine	0.5296	0.5745	0.5096	0.5518
11	Decision Tree	0.5291	0.5300	0.5378	0.5422
12	Gradient Boosting	0.5068	0.5401	0.5090	0.5353
13	Quadratic Discriminant Analysis	0.0771	0.1478	0.0994	0.4853

Table A.5: Performances of the top-5 classifiers after tuning on the validation set of Bio.

	Model	F1-score	Precision	Recall	Accuracy
0	Naive Bayes	0.6451	0.5063	0.9096	0.5142
1	Logistic Regression	0.5791	0.6069	0.5744	0.5878
2	Ridge Classifier	0.5753	0.6069	0.5660	0.5877
3	Linear Discriminant Analysis	0.5733	0.6065	0.5660	0.5838
4	Random Forest Classifier	0.6026	0.6204	0.6071	0.6087

Table A.6: Performances of the top-5 classifiers after tuning on the test set of Bio.

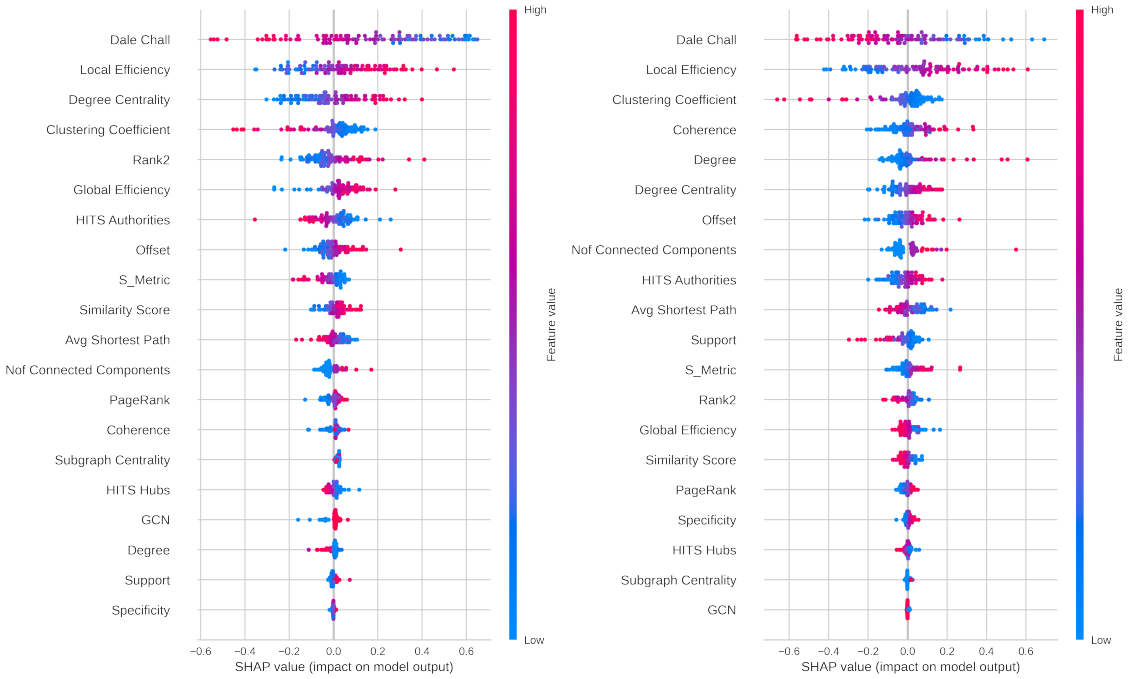
	Model	F1-score	Precision	Recall	Accuracy
0	Naive Bayes	0.6667	0.5	1.0	0.5048
1	Logistic Regression	0.6792	0.6667	0.6923	0.6762
2	Ridge Classifier	0.6667	0.6604	0.6731	0.6667
3	Linear Discriminant Analysis	0.6792	0.6667	0.6923	0.6762
4	Random Forest Classifier	0.5862	0.5312	0.6538	0.5429

Comparing the performances of all 14 classifiers in the multi-class setting on the validation set of Newsela using default hyperparameters (Table A.1) with the performances of the top-5 models after hyperparameter optimization (Table A.2) indicates that no overfitting occurred as the performances on the validation dataset are similar to those obtained on the test set as shown in Table A.3. The same observation about overfitting also holds for Bio as shown in Tables A.4, A.5, and A.6.

Appendix B

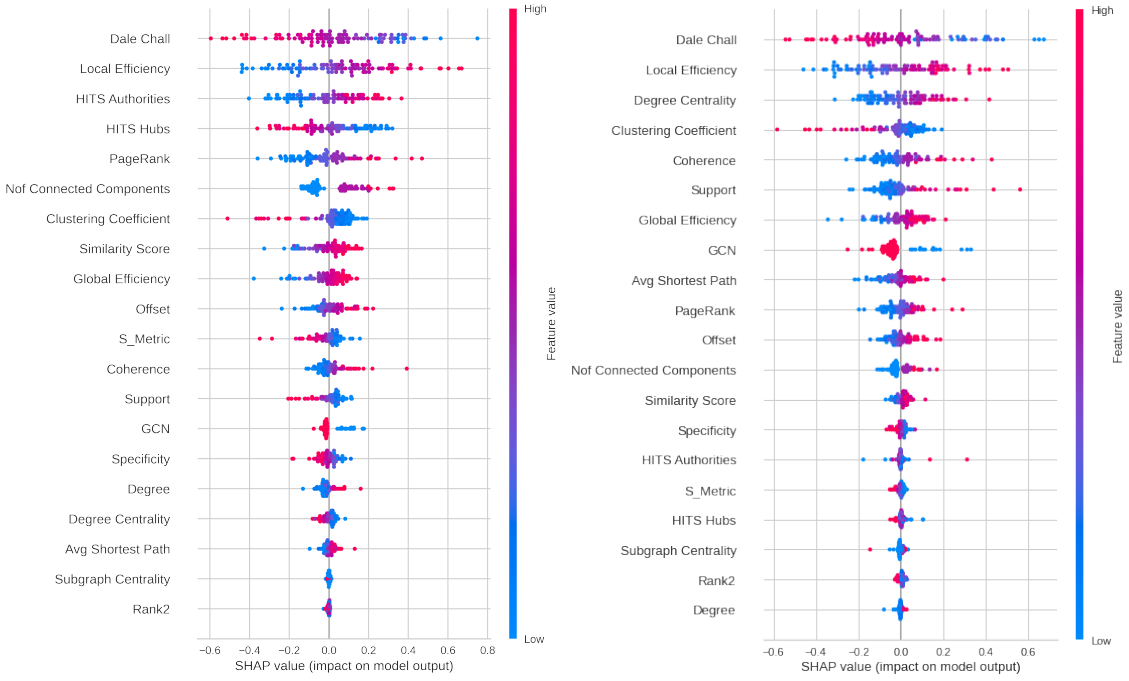
Instability of Bio Beeswarm Plots

As shown in Fig B.1, when splitting the available documents in Bio randomly into training and test set in four different ways, the resulting Beeswarm plots vary substantially in terms of the feature importance that is attributed to each feature. Thus, we do not draw any conclusions from Beeswarm plots in Bio and just report the results for reference.



(a) First split.

(b) Second split.



(c) Third split.

(d) Fourth split.

Figure B.1: Resulting Beeswarm plots when splitting Bio randomly into training and test set four times.

Bibliography

[1] C.-L. Tang, J. Liao, H.-C. Wang, C.-Y. Sung, and W.-C. Lin, "Conceptguide: Supporting online video learning with concept map-based recommendation of learning path," in *Proceedings of the Web Conference 2021*, 2021, pp. 2757–2768.

[2] J. Karreman, T. Van der Geest, and E. Buursink, "Accessible website content guidelines for users with intellectual disabilities," *Journal of applied research in intellectual disabilities*, vol. 20, no. 6, pp. 510–518, 2007.

[3] J. S. Chall and E. Dale, *Readability revisited: The new Dale-Chall readability formula*. Brookline Books, 1995.

[4] Y. Fujinuma and M. Hagiwara, "Semi-supervised joint estimation of word and document readability," in *Proceedings of the Fifteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-15)*, 2021, pp. 150–155.

[5] K. Collins-Thompson, "Computational assessment of text readability: A survey of current and future research," *ITL-International Journal of Applied Linguistics*, vol. 165, no. 2, pp. 97–135, 2014.

[6] S. Vajjala Balakrishna, "Analyzing text complexity and text simplification: Connecting linguistics, processing and educational applications," Ph.D. dissertation, Universität Tübingen, 2015.

[7] R. G. Benjamin, "Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty," *Educational Psychology Review*, vol. 24, no. 1, pp. 63–88, 2012.

[8] C. Initiative, "Common core standards for english language arts & literacy in history/social studies, science, and technical subjects," *Washington, DC: Council of Chief State School Officers (CCSSO). Retrieved January*, vol. 22, p. 2011, 2010.

[9] A. Bailin and A. Grafstein, "The linguistic assumptions underlying readability formulae: A critique," *Language & communication*, vol. 21, no. 3, pp. 285–301, 2001.

[10] S. Štajner and I. Hulpuș, "Automatic assessment of conceptual text complexity using knowledge graphs," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 318–330.

[11] I. Hulpuș, S. Štajner, and H. Stuckenschmidt, "A spreading activation framework for tracking conceptual complexity of texts," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 3878–3887.

[12] S. Štajner and I. Hulpuș, "When shallow is good enough: Automatic assessment of conceptual text complexity using shallow semantic features," in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 1414–1422.

[13] G. I. PS, J. Fadlil, R. C. HP, and H.-K. Pao, "Text comprehensiveness ranking," in *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 1. IEEE, 2015, pp. 21–25.

[14] R. DeKeyser, "Of moving targets and chameleons: Why the concept of difficulty is so hard to pin down," *Studies in Second Language Acquisition*, vol. 38, no. 2, pp. 353–363, 2016.

[15] R. Johnson, "Readability." *School Science Review*, vol. 60, no. 212, pp. 562–68, 1979.

[16] G. Pallotti, "A simple view of linguistic complexity," *Second Language Research*, vol. 31, no. 1, pp. 117–134, 2015.

[17] W. Xu, C. Callison-Burch, and C. Napoles, "Problems in current text sim-

plification research: New data can help," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 283–297, 2015.

[18] Z. Huang, Q. Liu, E. Chen, H. Zhao, M. Gao, S. Wei, Y. Su, and G. Hu, "Question difficulty prediction for reading problems in standard tests," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[19] Q. Li, S. Huang, Y. Hong, and S.-C. Zhu, "A competence-aware curriculum for visual concepts learning via question answering," in *European Conference on Computer Vision*. Springer, 2020, pp. 141–157.

[20] F. B. Baker, *The basics of item response theory*. ERIC Clearinghouse on Assessment and Evaluation, 2001.

[21] V. E. Venugopal and P. S. Kumar, "Difficulty-level modeling of ontology-based factual questions," *Semantic Web*, vol. 11, no. 6, pp. 1023–1036, 2020.

[22] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer *et al.*, "Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic web*, vol. 6, no. 2, pp. 167–195, 2015.

[23] T. François, "When readability meets computational linguistics: a new paradigm in readability," *Revue française de linguistique appliquée*, vol. 20, no. 2, pp. 79–97, 2015.

[24] W. H. DuBay, *The Principles of Readability*. ERIC Clearinghouse, 2004.

[25] J. Nelson, C. Perfetti, D. Liben, and M. Liben, "Measures of text difficulty: Testing their predictive value for grade levels and student performance," *Council of Chief State School Officers, Washington, DC*, 2012.

[26] K. M. Sheehan, M. Flor, and D. Napolitano, "A two-stage approach for generating unbiased estimates of text complexity," in *Proceedings of the Workshop on Natural Language Processing for Improving Textual Accessibility*, 2013, pp. 49–58.

[27] T. Deutsch, M. Jasbi, and S. Shieber, "Linguistic features for readability assessment," *arXiv preprint arXiv:2006.00377*, 2020.

[28] S. Vajjala and I. Lučić, "Onestopenglish corpus: A new corpus for automatic readability assessment and text simplification," in *Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications*, 2018, pp. 297–304.

[29] S. Vajjala and D. Meurers, "On improving the accuracy of readability classification using insights from second language acquisition," in *Proceedings of the seventh workshop on building educational applications using NLP*, 2012, pp. 163–173.

[30] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[31] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis, "Conditional variable importance for random forests," *BMC bioinformatics*, vol. 9, no. 1, pp. 1–11, 2008.

[32] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 2020, pp. 607–617.

[33] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[34] M. Harbach, S. Fahl, P. Yakovleva, and M. Smith, "Sorry, i don't get it: An analysis of warning message texts," in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 94–111.

[35] E. Dale and J. S. Chall, "The concept of readability," *Elementary English*, vol. 26, no. 1, pp. 19–26, 1949.

[36] M. L. Lewis and M. C. Frank, "The length of words reflects their conceptual complexity," *Cognition*, vol. 153, pp. 182–195, 2016.

[37] J.-P. Doignon and J.-C. Falmagne, "Spaces for the assessment of knowledge," *International journal of man-machine studies*, vol. 23, no. 2, pp. 175–196, 1985.

[38] M. Morsey, J. Lehmann, S. Auer, and A.-C. N. Ngomo, "Dbpedia sparql benchmark–performance assessment with real queries on real data," in *International semantic web conference*. Springer, 2011, pp. 454–469.

[39] A. Bougouin, F. Boudin, and B. Daille, "Topicrank: Graph-based topic ranking for keyphrase extraction," in *International joint conference on natural language processing (IJCNLP)*, 2013, pp. 543–551.

[40] F. Boudin, "pke: an open source python-based keyphrase extraction toolkit," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, Osaka, Japan, December 2016, pp. 69–73. [Online]. Available: http://aclweb.org/anthology/C16-2015

[41] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "Dbpedia spotlight: shedding light on the web of documents," in *Proceedings of the 7th international conference on semantic systems*, 2011, pp. 1–8.

[42] I. Hulpus, C. Hayes, M. Karnstedt, and D. Greene, "Unsupervised graph-based topic labelling using dbpedia," in *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013, pp. 465–474.

[43] A. Olieman, H. Azarbonyad, M. Dehghani, J. Kamps, and M. Marx, "Entity linking by focusing dbpedia candidate entities," in *Proceedings of the first international workshop on Entity recognition & disambiguation*, 2014, pp. 13–24.

[44] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world'networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[45] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web." Stanford InfoLab, Tech. Rep., 1999.

[46] J. Hopcroft and R. Tarjan, "Algorithm 447: efficient algorithms for graph manipulation," *Communications of the ACM*, vol. 16, no. 6, pp. 372–378, 1973.

[47] V. Latora and M. Marchiori, "Efficient behavior of small-world networks," *Physical review letters*, vol. 87, no. 19, p. 198701, 2001.

[48] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[49] E. Estrada and J. A. Rodriguez-Velazquez, "Subgraph centrality in complex networks," *Physical Review E*, vol. 71, no. 5, p. 056103, 2005.

[50] I. Beichl and B. Cloteaux, "Measuring the effectiveness of the s-metric to produce better network models," in *2008 Winter Simulation Conference*. IEEE, 2008, pp. 1020–1028.

[51] J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes, "Improving efficiency and accuracy in multilingual entity extraction," in *Proceedings of the 9th international conference on semantic systems*, 2013, pp. 121–124.

[52] C. of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division, *Common European framework of reference for languages: Learning, teaching, assessment*. Cambridge University Press, 2001.

[53] M. Xia, E. Kochmar, and T. Briscoe, "Text readability assessment for second language learners," in *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*. San Diego, CA: Association for Computational Linguistics, Jun. 2016, pp. 12–22. [Online]. Available: https://aclanthology.org/W16-0502

[54] M. Negishi, T. Takada, and Y. Tono, "A progress report on the development of the cefr-j," in *Exploring language frameworks: Proceedings of the ALTE Kraków Conference*, 2013, pp. 135–163.

[55] A. J. Stenner, H. Burdick, E. E. Sanford, and D. S. Burdick, "How accurate are lexile text measures?" *Journal of Applied Measurement*, vol. 7, no. 3, p. 307, 2006.

[56] Z. Zhu, D. Bernhard, and I. Gurevych, "A monolingual tree-based translation model for sentence simplification," in *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, 2010, pp. 1353–1361.

[57] J. Milton and T. Alexiou, "Vocabulary size and the common european framework of reference for languages," in *Vocabulary studies in first and second language acquisition*. Springer, 2009, pp. 194–211.

[58] M.-S. Paukkeri, M. Ollikainen, and T. Honkela, "Assessing user-specific difficulty of documents," *Information Processing & Management*, vol. 49, no. 1, pp. 198–212, 2013.