Difficulty of Items – Predictions on Linguistic Features

Anna Winklerová

Masaryk University, Botanická 68a, 60200, Brno, Czechia

Abstract

To fulfill adaptive and mastery learning parameters of an educational system (both learning and assessment), it is necessary to continuously develop and manage a large item pool containing thousands of items in a properly designed structure. Content management can be efficiently supported by utilizing augmented intelligence models that can deduce behaviour of items in the system based on linguistic features, independent on user data.

This paper focuses on categorizing linguistic features for short L2 English multiple choice items, discusses ways of feature selection towards feature interpretability and its consequences on model prediction. It demonstrates practical application of prediction results for item management and further meaningful feature development.

Keywords

Item difficulty prediction, Second language acquisition, Natural language feature engineering, Interpretable features, Estimation of question statistics

1. Introduction

The overall accuracy behind an adaptive educational (both learning and assessment) system performance is dependent on student – item interactions.Difficulty of an item is one of the most descriptive metrics of how an item behaves in the system but the reasoning for the behaviour is more complex and tricky. For instance, we can easily identify that an item behaves differently than expected by measuring the item's error rate distance from the mean error rate of the whole set of items however the reasoning for this behaviour can not be easily done without further complementary data. This applies to the item complexity features that are free of student interaction [1], in other words, textual and form related content of items. Analysis of linguistic features characterizing item complexity in combination with item difficulty serve as a valuable insight into item behavior in the system.

Features engineered from even such short digital entities as multiple choice gap filling (MCQ) items in educational content summarized in recent research count in hundreds. Linguistic features free of user interaction data take a vast share. They diverge from simple lexical and surface features, through syntax and basic semantic features to composite discourse and embedding features. Current acceleration in computational linguistics brings profits in improved methods and tools for feature engineering but also challenges in the dimensionality reduction and proper machine learning model application to reach practical goals such as difficulty prediction.

EvalLAC'24: Workshop on Automatic Evaluation of Learning and Assessment Content, July 08, 2024, Recife, Brazil

anna.winklerova@mail.muni.cz (A. Winklerová)

Our research is motivated mainly by two groups of users: content decision makers and model developers. These two distinctive groups of users represent different views on item features as they both stress out different objectives. Decision makers are mostly domain specialists and they need to comprehend the results of the predictions and its underlying decisions. On the other hand model developers aim to improve the precision of the predictions no matter how complex and inarticulate the models and features are. Based on the work of Alexandra Zytek et al. [2] we examine the overlapping *Interpretable feature space* and *Model-ready feature space* to find the suitable set of feature properties to select relevant interpretable features in the difficulty prediction setting.

The aim of this paper is to (1) describe feature engineering methods in the context of item difficulty prediction (2) report on the ongoing research of automated methods for interpretable feature selection methods on the 230 feature set of real life data from a educational system containing thousands of items on English L2 practice with thousands of student interactions and to (3) demonstrate exploitation of augmented intelligence for decision making in item management.

The difficulty estimation is implemented as a regression task utilizing simple Random Forest (RF) ensemble algorithm and Gradient Boosting Trees (GBT) for comparison. The focus of this work is not on optimizing the ML algorithms, and cross validation of hyperparameter setting was not performed. Rather, the ML algorithms are used in static setting for comparable results on feature engineering and selection.

2. Relevant research

The recent comprehensive state-of-the-art overviews on item difficulty prediction [3, 4] have demonstrated intensive ongoing research in various learning contexts. The predominant reason to predict difficulty of an item is to accelerate establishment of new items in educational systems, which could reduce the resource cost of item pool management and administration. Behind the pursue of these clear quantitative goals there also emerge data quality benefits of item feature modeling. As mentioned in[5], the data-driven insights, such as intensive item modeling with linguistic features, can (1) inform on overall structuring of content and in particular (2) help predict the individual learners' difficulties and skills.

Distilling features as numerical representations of textual parameters of a natural language text passage is increasingly influential in recent research and applications. There are being developed libraries for basic handcrafted features engineering such as LFTK [6] containing 220 features covering lexical, semantic, syntactic or discourse features. The efforts to categorize and standardize text features are creditable, reduce the heavy lifting of text preprocessing and, in our research context, enable systematically build up specific features. In addition, implementing standardized vector representations of items by linguistic features can contribute to sharing data in educational and assessment systems. Lack of data for methodology comparison is identified as one of the obstacles in item difficulty prediction research.

With the increasing number of item features and the aim of practical usage of the predictive models, the need for interpretable features is eminent. Consistently with the recent work of Alexandra Zytek et al. [2] we define the key stakeholders in item pool management as *decision* *makers* and *model developers*. Decision makers use the model results to gain insight on creating new items, identifying items for revision and taking appropriate action such as modifying or removing dysfunctional items, inspecting the coherency of domain subsets (e.g., splitting topics, adding new topics), analysing item functioning in order to gain insight with applicable actions [7]. These users need the model results to be understandable and consistent with their domain expertise.

Model developers use and finetune machine learning algorithms to improve model precision for a given task (such as difficulty prediction) on a particular dataset. Their motivation is therefore focused on collecting predictive, model-ready features correlating with the target variable.

Building models on interpretable features aims to build trust [8] in end users and opens door to further development including crowd feature engineering. With compliance with [2] we require the interpretable features to have at the same time the following properties, or have clearly defined transform functions from interpretable to model-ready feature space. Relevant interpretable properties have features that are:

- *Understandable* – related to real-world metric, e.g. *Age of acquisition* stated in age rather than given by a scale from 1 to 100,
- *Readable* – labeled with plain human language with understandable meaning (for our purpose *readable* is defined to the extent of *human-worded* features), e.g. *Number of words in a sentence* instead of *Average number of tokens per sentence*,
- *Meaningful* – with clear relation to the target variable, comprehensible to decision makers,

From the model point of view, a feature must be *predictive*: improve a model performance, have sufficient data coverage, explain data variance and be independent on other features.

Interpretable features subset from the set of 230 features needs to be justified on a computational basis. Based on comprehensive review of dimensionality reduction techniques by R. Zebari et al. [9] we examine two main groups of approaches of achieving reduced features set by *feature selection* and *feature extraction.* Although we have experimented with feature extraction methods such as PCA on feature correlation clusters obtaining significant computational time improvements while keeping the prediction accuracy, the main objective of our work lies heavily in the interpretability of features' significance to item modeling. Therefore, the main ongoing work leverages from the feature selection methods.

The studied feature selection mechanisms focus on maximizing relevant information while minimizing redundant information. The research in this area deals with automated calculation of minimal or optimal number of features that still cover the relevant variance of data while decreasing bias or noise, suggesting various approaches and methods such as mutual information, vector variance inflation, clustering or correlation based analysis.[10, 11]

Although, in many cases of feature selection, the best performance is obtained by using all available features, [12] improvement in item difficulty prediction by feature subset selection was demonstrated in [5]. Furthermore, models with smaller interpretable set of features are more usable to decision makers and are open to further development by users that are not machine learner experts.

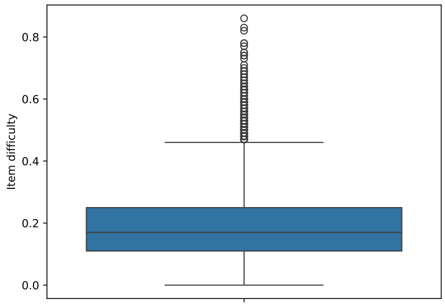

Figure 1: Distribution of difficulty.

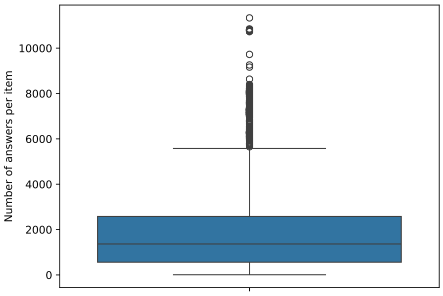

Figure 2: Number of answers distribution.

Table 1
Examples of items.

Question	Correct answer	Distractor	Difficulty
How many days ____ after school?	have you been running	have you run	0.37
____ the morning	In	At	0.26
They hope ____ us next year.	to visit	visiting	0.21
Hey! Check this ____!	out	in	0.17
You should cut ____ on sugar.	back	over	0.42

3. Framework implementation

Our work does not focus merely on item difficulty prediction, but on a wider context of augmented intelligence models supporting decisions towards item pool management. Secondly, we aim to give insights and recommendations to model developers concerning (1) which feature types are still meaningful to further investigate in the context of educational content and also (2) which features are useful in a given ML application task. Therefore, the individual pipeline steps starting with the text preprocessing up to result evaluation are covered in a framework.

In this paper, we focus mainly on the implementation and evaluation of the feature engineering and selection methods.

3.1. Umíme dataset

The items in this dataset are from private educational system for practicing grammar, vocabulary and use of English. It is targeted on L2 learners of English from the first years of studying language up to advanced high school learners. Over 5900 items are structured in 34 resource sets focusing on different concepts of language. Difficulty of items is calculated as an error rate on the scale from 0 to 1.

3.2. Feature engineering

MCQ items are short, one to two sentence long passages of text. The lexical and semantic representation of these sentences hardly contains every aspect of what makes an item difficult. To distinguish MCQ relevant features, we propose describing item characteristics as *static* or *dynamic*. Imagine a sentence *The United States have around 330 million inhabitants.* This sentence consists of static features including POS tags, syntax tree parameters, surface features such as word syllables count or even various metrics for readability indexes, word frequencies or age of acquisition.

Next assume MCQ item created from the basic sentence. The item has one correct answer (stated as the first in the bracket) and one distractor. The item can be created as one of the following examples:

- The United States have around [330;660] million inhabitants.
- The United States [has;have] around 330 million inhabitants.
- The United States has around 330 million[s;_] inhabitants.

Although the static features of the above-mentioned MCQ items are almost identical, the dynamic of student engagement in the individual items is essentially different. Features describing the dynamic item component can be derived from the item answers or grammar pattern of underlying knowledge domain. These features try to explain a context-sensitive stimuli leading to a student action and resolving into item difficulty.

Results of our experiments show that the development of dynamic item features contributes to the difficulty prediction and item behaviour modeling in an educational system.

3.2.1. Feature set

After a standard text processing steps containing item purification, contraction expansion (i.e. it's – it is), tokenization, stopword filtering etc. we derived numerical representation of handcrafted features provided by the LFTK package for Python and handcrafted more features describing mostly the dynamic item component.

In the LFTK package, there are currently 220 features divided into overlapping sections of foundation (e.g. verb count) and derivation (e.g. average verb count per word/per sentence) features from diverse domains and families (syntactic, semantic, discourse or named entities). The short, mostly one sentence long items, do not utilize all features from the LFTK package as many of them are designed for longer passages of text (readability measures, counts of unique words per sentence).

MCQ specific features are not present in the package. The 10 remaining features were derived from textual parts of items using nlp libraries such as SpaCy, nltk for distance metrics for distractor similarity measures or CEFR level dictionary for vocabulary difficulty analysis. These features are described in table 2

The resulting size of features set is 230 containing all LFTK features and basic MC specific features.

3.3. Dimensionality reduction

In order to achieve model interpretability and usability, we performed several feature reduction experiments based on different methods and underlying calculations. To lower the high number of dimensions and to reduce bias caused collinearity of features, we have experimented with (1) hierarchical clustering based on feature correlations and Principal Component Analysis as well as (2) various approaches for feature selection.

Table 2
List of MCQ specific handcrafted features in model-ready description.

Handcrafted feature name	Description
Manhattan vector distance	Difference between correct and wrong sentence represented by LFTK feature vector.
Distractor similarity	Edit distance between the correct answer and distractor.
Sentence similarity	Edit distance between the whole correct sentence (correct answer inputed into the gap) and wrong sentence.
Certainty	BERT fill in the mask pretrained model percentual value of certainty for masked expression.
Gap position	Normalized score of the position of the gap in the sentence on scale 0-1 representing beginning and end of sentence respectively.
Q_0 mean CEFR	Mean of all words in CEFR A1-C2 label on scale 1-6
Q_0 max CEFR	Mean of all words in CEFR A1-C2 label on scale 1-6
Distractor max CEFR	Distractor max CEFR A1-C2 label on scale 1-6
Correct max CEFR	Correct answer max CEFR A1-C2 label on scale 1-6
Average sentence parser tree depth	Sentence parse depth represented by average number of children of parse tree nodes.

3.3.1. Feature extraction

The feature extraction methods such as PCA are very straightforward and automatically applicable on any data types (with respect to numerical representations and normalization), preserving data variation and improving ML tasks in some cases. Instead of applying PCA across the whole set of 230 features, we first implemented clustering on correlated features as illustrated in figure **??**. The figure shows large groups of highly correlated linguistic features (e.g. token absolute and average lengths and counts). Large clusters with closely similar features (correlating more than 0.8) indicate that these types of features are not interesting for further investigation, as the informational gain has already been exhausted. These features can be substituted in further computations by representative selection or linear combination (e.g. PCA).

On the other hand, the low size clusters may contain features of significant importance towards the ML task and represent promising areas for deeper feature analysis and engineering (e.g. distractor similarity).

The size of clusters is parametrized by the distance factor which is directly influencing the number and size of feature groups, PCA variation ratio and difficulty prediction accuracy.

In this particular experiment, PCA was applied on features from clusters containing at least 5 features. Features in smaller clusters are used as individual independent variables in the training set.

The extracted principal component features lose their interpretable qualities and can not be used unless a transformation function is applied which translates the principal components or factors into an abstract concept. Such abstract concept (could be labeled for example as textual complexity) must be understandable and meaningful to the decision maker.

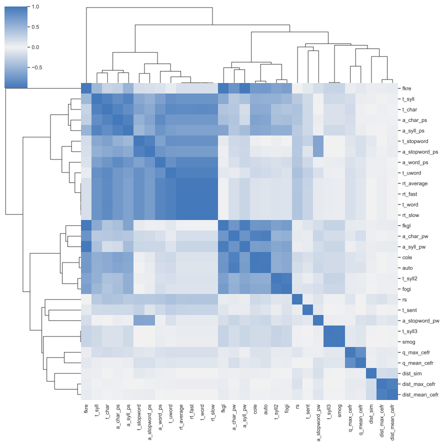

Figure 3: Hierarchical clustering based on pairwise correlation.

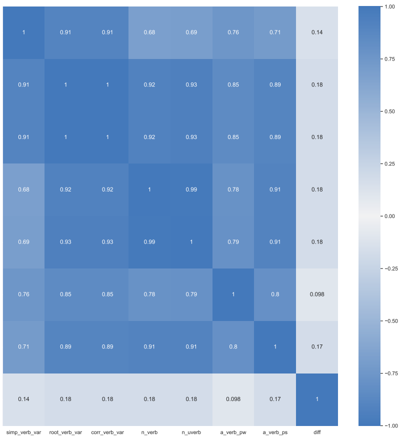

Figure 4: Correlation cluster of features based on verbs and their correlation with difficulty.

3.3.2. Feature selection

To investigate the methods of feature selection that best fit our dataset and ML task, we made use of the fine granularity of the dataset structure. The feature correlations with target value, but also collinearity among features proved to be very diverse among different item resource sets. Future work on the item pool and feature set will focus on finding general characteristics in resource sets with low prediction accuracy. The desired result is a predictive model that is as general as possible yet able to explain most of the variance in various data.

Two approaches are described in the following text in detail: features selected based on model-feature importances and interpretable features selected based on hierarchical clustering.

Features selected based on model-feature importances. This approach combines results of the random forest decision algorithm and the model feature importances and structure of the data. The difficulty prediction was calculated separately on each resource set (5900 items in 34 resource sets). The top 20 features in each resource set were multiplicated by the importance of each given feature calculated by the RF and further multiplicated by the Pearson's correlation coefficient of the prediction on the whole resource set. From this cumulated importances, 10 features were selected and new prediction was performed.

Interpretable features selected based on hierarchical clustering. This approach uses the results of hierarchical clustering depicted in figure 3.3.1 and selects (manually) representatives of the constructed clusters that are predictive and interpretable. For instance all features from the LFTK package based on verbs ended up in the "verb" cluster as shown in figure 4. From the features in the cluster, we have selected the most interpretable feature while maximizing the predictive quality (correlation with target value). These are usually translated to an understandable definition and were created as combination of model-ready features.

The table 5 shows different results of predictions on structured data. Table contains worst and best predictions expressed with the pearson correlation. The difficulty predictions vary greatly among different resource sets.

The overall accuracy of predictions on all data combined is stated in table 6.

Table 3
Features selected based on the cumulative RF feature importance across all RS.

Feature	Source	Type
Manhattan vector distance	handcrafted	dynamic
Distractor similarity (correct answer, distractor string similarity)	handcrafted	dynamic
Kuperman AoA per word	LFTK	static
SubtlexUS	LFTK	static
BERT fill in the mask pretrained certainty	handcrafted	dynamic
Gap position	handcrafted	dynamic
Question mean CEFR A1–C2 class	handcrafted	static
Sentence similarity (correct vs wrong sentence)	handcrafted	dynamic
Average count of auxiliary words per word	LFTK	static
Correct answer max CEFR A1–C2 class	handcrafted	static
Average punctuation per word	handcrafted	static
Average verb per word	LFTK	static

Table 4
Interpretable features from feature clusters.

Feature representation	Underlying feature	Source
Item length	Word count	LFTK
Gap position	Normalized gap position	handcrafted
Gap expectation	BERT fill in the mask pretrained probability	handcrafted
Similarity of options	Edit distance correct vs wrong sentence	handcrafted
Consistency of vocabulary	$CEFRmaxQ_c - CEFRmeanQ_c$	handcrafted
Readability	Flesch Reading Ease	LFTK
Age of Acquisition	Average Kuperman age of acquistion of words per word	
Usual vocabulary	SubtlexUS word frequency	LFTK
Complexity of sentence	Average sentence parser tree depth	handcrafted

4. Applicable results for decision making

Table 7 shows practical steps in evaluation of difficulty prediction results. Difficulty of the item *"My sisters ____ a beach house."* based on linguistic features was predicted 0.24, whereas the observed difficulty (error rate) is calculated as 0.6.

With further investigation of linguistic parameters of similar items within the same resource set (item modeling in context), the table 8 shows that items with similar mask *"s have"* represent an outlier with suggested action for the content developer to add more similar items in order to balance structure and content of the resource set. The overall number of items with a correct answer containing *have got* or *has got* from the whole resource set is 67 of which only two represent similar token linkage with plural noun *My sisters* or *My cousins* respectively. The token *(grand)parents* implicates plural form of the verb *have got.* Other items use different subject forms, such as plural pronouns *they, we, you* and multiple personal names *Jane and Pete.*

This item could have been marked as dysfunctional by simply comparing the outlying difficulty towards the mean difficulty of the resource set. However, the further analysis leading to content developer action is heavily dependent on rich item modeling (POS tags and syntactic

Table 5
Comparison of predictions for different feature sets on structured data.

RS name	RF import.	rP	RF230	rP	RF interpr. rP
Present perfect: simple vs. continuous	1	0.01	1	-0.01	0.08
Phrasal verbs	2	0.03	11	0.28	0.27
Wh- question	3	0.06	3	0.1	0.2
Some, any, no, every	4	0.08	2	0.06	0.11
Possessive pronouns	30	0.57	31	0.61	0.53
Present tense: negatives	31	0.66	30	0.6	0.59
To do, to have, to be in past simple	32	0.68	32	0.69	0.61
Can vs. could	33	0.75	33	0.69	0.65
To be in present simple	34	0.75	34	0.75	0.73

Table 6
Comparison of predictions on different feature sets for on all data (5900 items).

Prediction model	Pearson	RMSE
All features RF	0.394	
Top 10 RF importance features	0.347	
Top 10 interpretable features RF	0.313	

Table 7
Results on prediction outlier with the highest difference between observed and predicted difficulty – error rate. TER = True error rate, PER = Predicted error rate.

Item text	Correct option	Distractor	TER	PER
My sisters ____ a beach house.	have got	has got	0.6	0.24
My cousins ____ a farm in Oregon.	have got	has got	0.46	0.26
Tourists ____ their cameras ready.	have got	has got	0.19	0.23
My parents ____ two dogs and a cat.	have got	has got	0.17	0.19
My grandparents ____ a big garden.	have got	has got	0.08	0.10

marking).

The other side of the difficulty prediction error scale contains the cases of significantly higher predicted values than true values. This is a much more interesting result considering the novelty insight into data. The explanation is often in an unintentional clue in the item revealed by the students, or wrongly placed item (too easy in the context of other items, but not necessarily the easiest). This type of item deficiency can be corrected by the content developer, who should be able to find the hidden clue and rewrite the item, or place it in different set of items.

In table 9 is an example of an item with the highest difference between predicted difficulty and true difficulty. The explanation lies in markedly different options. Correct answer *Do I live* and distractor *Have I live* are a combination that has not appeared in any other item and the distractor is of no attraction for the student.

Table 8
Analysis of items with *have/has got* grammar focus within the resource set *To do, to have, to be in present simple*, with average difficulty: 0.16.

Number of items	Form	Example items	Error rate
37	have got	We have got a dog.	0.15
		They have got two daughters.	0.18
		Ann and Bill have got a new house.	0.10
30	has got	My father has got two cars.	0.20
		My dog has got long ears.	0.20
		Ela has got long hair.	0.13

Table 9
Example of an item easier than predicted.

Item text	Correct option	Distractor
_____ in a city?	Do I live	Have I live
_____ he have an apple?	Does	Do
_____ she there?	Is	Are
_____ they sisters?	Are	Is
_____ he need our help?	Does	Do

5. Conclusion

Exploiting recent advancements in linguistic computations and natural language processing, this paper demonstrates a framework covering steps of feature engineering, dimensionality reduction and practical application. The main importance is in the interpretability of model by its features. We believe that models with understandable features are of the most use to decision makers. Furthermore, the interpretability can lead to model precision improvement as more involved users, who are not ML specialists, can contribute to the dynamic feature engineering.

Difficulty prediction tasks give deeper insight into the anatomy of items in context of student skills which helps to manage the content of educational systems.

References

[1] R. Pelánek, T. Effenberger, J. Čechák, Complexity and difficulty of items in learning systems, International Journal of Artificial Intelligence in Education 32 (2022) 196–232.

[2] A. Zytek, I. Arnaldo, D. Liu, L. Berti-Equille, K. Veeramachaneni, The need for interpretable features: Motivation and taxonomy, ACM SIGKDD Explorations Newsletter 24 (2022) 1–13.

[3] S. AlKhuzaey, et al., Text-based question difficulty prediction: A systematic review of automatic approaches, 2023.

[4] L. Benedetto, et al., A survey on recent approaches to question difficulty estimation from text, 2023.

[5] I. Pandarova, T. Schmidt, J. Hartig, A. Boubekki, R. D. Jones, U. Brefeld, Predicting the

difficulty of exercise items for dynamic difficulty adaptation in adaptive language tutoring, 2019.

[6] B. W. Lee, J. H.-J. Lee, Lftk: Handcrafted features in computational linguistics, 2023.

[7] R. Pelánek, T. Effenberger, A. Kukučka, et al., Towards design-loop adaptivity: identifying items for revision, Journal of Educational Data Mining 14 (2022) 1–25.

[8] S. R. HONG, J. HULLMAN, E. BERTINI, Human factors in model interpretability: Industry practices, challenges, and needs, arXiv preprint arXiv:2004.11440 (2020).

[9] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, J. Saeed, A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction, 2020.

[10] H. Zhou, X. Wang, R. Zhu, Feature selection based on mutual information with correlation coefficient, Applied Intelligence 52 (2022) 5457–5474.

[11] H. Liu, Z. Wu, X. Zhang, Feature selection based on data clustering, in: Intelligent Computing Theories and Methodologies: 11th International Conference, ICIC 2015, Fuzhou, China, August 20-23, 2015, Proceedings, Part I 11, Springer, 2015, pp. 227–236.

[12] M. A. Munson, R. Caruana, On feature selection, bias-variance, and bagging, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2009, pp. 144–159.

6. Online Resources

- The Umime educational system umimeto.org
- LFTK lftk.readthedocs.io,
- scikit scikit-learn.org,
- BERT Fill-Mask pretrained huggingface.co/google-bert.