Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 6812–6818
Marseille, 11–16 May 2020

Predicting Item Survival for Multiple Choice Questions in a High-stakes Medical Exam

Victoria Yaneva[1], **Le An Ha**[2], **Peter Baldwin**[1], **Janet Mee**[1]
1 - National Board of Medical Examiners, Philadelphia, USA
2 - Research Institute in Information and Language Retrieval, University of Wolverhampton, UK
{vyaneva, pbaldwin, jmee}@nbme.org; l.a.ha@wlv.ac.uk

Abstract

One of the most resource-intensive problems in the educational testing industry relates to ensuring that newly-developed exam questions can adequately distinguish between students of high and low ability. The current practice for obtaining this information is the costly procedure of pretesting: new items are administered to test-takers and then the items that are too easy or too difficult are discarded. This paper presents the first study towards automatic prediction of an item's probability to "survive" pretesting (item survival), focusing on human-produced MCQs for a medical exam. Survival is modeled through a number of linguistic features and embedding types, as well as features inspired by information retrieval. The approach shows promising first results for this challenging new application and for modeling the difficulty of expert-knowledge questions.

Keywords: Multiple Choice Questions, Difficulty Prediction, Educational Applications

1. Introduction

Large-scale testing relies on a pool of test questions, which must be replenished, updated, and expanded over time[1]. Writing high-quality test questions is challenging as they must satisfy certain quality standards before they can be used to score examinees. These standards are based on statistical criteria and ensure that: i) items are not too easy or too difficult for the intended examinee population, and ii) the probability of success on each item is positively related to overall examinee performance (Section 3.). While the exact thresholds vary, most exam programs have such a requirement. Even when item writers are well-trained and adhere to industry best practices, it has generally not been possible to identify which items will satisfy the various statistical criteria without first obtaining examinee responses through pretesting. Pretesting involves embedding new items within a standard live exam and, based on the collected responses, a determination is made about whether or not a given item satisfies conditions i) and ii). Items that meet the criteria are considered to have "survived" pretesting and can later be used to score examinees. The proportion of surviving items varies across programs; however, Brennan (2006) recommends pretesting at least twice the number of items needed.

While necessary, the enterprise of pretesting is costly. Scored items compete with pretest items for exam space, the scarcity of which can create a bottleneck. As a result, it is sometimes not possible to pretest as many new items as needed and some exam programs may not be able to afford pretesting at all. This problem is expected to grow with advances in automatic question generation (Gierl et al., 2018), where a large amount of new questions are generated but there is no criteria on how to evaluate their suitability for live use. Conceivably, having advance knowledge of an item's probability to survive can allow using the available pretesting slots for items that are more likely to pass the thresholds. To address these issues, we present a method for modeling item survival within a large-scale real-world data set of multiple choice questions (MCQs) for a high-stakes medical exam.

Contributions: i) The paper introduces a new practical application area of NLP related to predicting item survival for improving high-stakes exams. ii) The developed models outperform three baselines with a statistically significant difference, including a strong baseline of 113 linguistic features. iii) Owing to the generic nature of the features, the presented approach is generalizable to other MCQ-based exams. iv) We make our code available[2] at: `https://bit.ly/2EaTFNN`.

2. Related Work

Predicting item survival from item text is a new application area for NLP and, to the best of our knowledge, there is no prior work investigating this specific issue. The problem is, however, related to the limited available research on predicting question difficulty with the important difference that predicting survival involves predicting an additional item parameter that captures the relation between the probability of success for the individual item and overall examinee performance (Section 3.).

With regards to estimating question difficulty for humans, the majority of studies focus on applying readability metrics to language comprehension tests, where the comprehension questions refer to a given piece of text and, therefore, there is a relationship between the difficulty of the two (Huang et al., 2017; Loukina et al., 2016). For example, Loukina et al. (2016) investigate the extent to which the difficulty of listening items in an English language proficiency test can be predicted by the textual properties of the prompt by using text complexity features (e.g. syntactic complexity, cohesion, academic vocabulary, etc). In another study, Beinborn et al. (2015) rank the suitability

[1]This constant need for new test questions arises as the population of test-takers grows, new topics for exam content are identified, item exposure threatens exam security, etc.

[2]The questions cannot be released because of test security.

A 55-year-old woman with small cell carcinoma of the lung is admitted to the hospital to undergo chemotherapy. Six days after treatment is started, she develops a temperature of 38C (100.4F). Physical examination shows no other abnormalities. Laboratory studies show a leukocyte count of 100/mm3 (5% segmented neutrophils and 95% lymphocytes). Which of the following is the most appropriate pharmacotherapy to increase this patient's leukocyte count?	
(A) Darbepoetin	(B) Dexamethasone
(C) Filgrastim	(D) Interferon alfa
(E) Interleukin-2 (IL-2)	(F) Leucovorin

Table 1: An example of a practice item

and complexity of individual words as candidates for a fill-in-the-blanks test and this ranking is used to estimate the difficulty of the particular example. A slightly different approach to predicting test difficulty is presented in Padó (2017), where each question is manually annotated and labelled with the cognitive activities and knowledge necessary to answer it based on Bloom's Taxonomy of Educational Objectives (Bloom and others, 1956). The results indicate that questions that are low in Bloom's hierarchy of skills are easier to answer than ones high in the hierarchy. Nadeem and Ostendorf (2017) approach the same problem in an opposite way, where they aim to predict the skills required to solve assessment questions using a convolutional neural network (CNN). The ultimate goal of their experiments is to use annotated data with labels of such skills in order to automatically populate a Q-matrix of skills used in education to determine how questions should be graded (e.g., more points should be awarded for solving questions that require more skill).

Alsubait et al. (2013) show that the difficulty of newly generated questions can be manipulated by changing the similarity between item components, e.g. the distractors and the correct answer, the question and the distractors, the question and the correct answer, etc. This assumption is later on used by Ha and Yaneva (2018) in automatic distractor generation for multiple choice questions, where the system can rank distractors based on various similarity metrics.

In our prior work we predict MCQ difficulty and mean response times using a large number of linguistic features, in addition to embeddings (Ha et al., 2019; Baldwin et al., 2020). The results presented in Ha et al. (2019) show that the proposed approach predicts the difficulty of the questions with a statistically significant improvement over several baselines. As will be seen in Section 4., we use the full list of linguistic features to obtain a strong baseline prediction for item survival. More details on the individual features and their explanations can be found in Section 4..

3. Data

Data comprises 5,918 pretested MCQs from the Clinical Knowledge component of the United States Medical Licensing Examination (USMLE ®). An example of a test item is shown in Table 1. The part describing the case is referred to as *stem* and the incorrect answer options are known as *distractors*. All items tested medical knowledge and were written by experienced item-writers following a set of guidelines, stipulating adherence to a standard structure. These guidelines required avoidance of "window dressing" (extraneous material not needed to answer the item), "red herrings" (information designed to mislead the test-taker), and grammatical cues (e.g., correct answers that are longer or more specific than the other options). Item writers had to ensure that the produced items did not have flaws related to various aspects of validity. For example, flaws related to irrelevant difficulty include: *Stems or options are overly long or complicated, Numeric data not stated consistently* and *Language or structure of the options is not homogeneous*. Flaws related to "testwiseness" are: *Grammatical cues; The correct answer is longer, more specific, or more complete than the other options;* and *A word or phrase is included both in the stem and in the correct answer*. The goal of standardizing items in this manner is to produce items that vary in their difficulty and discriminating power due only to differences in the medical content they assess.

The items were administered within a standard nine-hour exam, and test-takers had no way of knowing that they would not be scored on these items. Each nine-hour exam contained approximately 40 pretest items and the data was collected through embedding the items in different live exam forms for four consecutive years (2012 - 2015). On average, each item was answered by 328 examinees ($SD = 67.17$). Examinees were medical students from accredited[3] US and Canadian medical schools taking the exam for the first time as part of a multistep examination sequence required for medical licensure in the US.

To survive, items had to satisfy two criteria:

- A proportion of correct answers between .30 and .95, i.e., the item had to be answered correctly by no fewer than 30% and no more than 95% of test-takers. Within the educational-testing literature, this proportion of correct answers is commonly referred to as a *P-value*. We adopt this convention here but care should be taken not to confuse this usage with a p-value indicating statistical significance. The P-value is calculated in the following way:

$$P_i = \frac{\sum_{n=1}^{N} U_n}{N},$$

[3] Accredited by the Liaison Committee on Medical Education (LCME).

where *Pi* is the p-value for item *i*, *Un* is the 0-1 score (correct-incorrect) on item *i* earned by examinee *n* , and *N* is the total number of examinees in the sample.

- The correlation (referred to as *Rb*) between examinees' responses on the given item and examinees' total test score had to be greater than zero. In other words, examinees who perform better on the test overall, must be more likely to succeed on the item than those examinees who performed worse. This ensures that the item, if used for scoring, will improve the quality of the total test score.

In our sample, half of the items (2,959) are from the surviving class, while the other 2,959 are from the non-surviving class. This distribution is in line with the observation by Brennan (2006) that the pretested items should be at least twice the number of items needed. From the non-surviving items, only 22% violated the Rb criterion, while 78% violated the P-value criterion.

4. Features

To model item survival (through P and Rb), we are concerned with several types of item characteristics. First, following (Ha et al., 2019), we extract 113 linguistic features accounting for several levels of linguistic complexity in order to understand the extent to which item survival can be attributed simply to the way the items are written. Such an understanding is useful in two main ways. First, it represents a strong baseline that other models can be compared to and second, it allows examining whether the item writers successfully minimize text complexity variation (the latter has a high practical value for test development). As a step beyond linguistic complexity, we explore a deeper-level semantic characteristics of the items by generating two types of embeddings presented below.

While these two approaches (linguistic features and embeddings) are widely used in various NLP applications, they do not account for the type of difficulty that is related to MCQs in particular, e.g., the relationship between item components. For this reason we develop a set of features inspired by automatic question answering (Ha and Yaneva, 2019), the aim of which is to quantify the difficulty of solving an MCQ for an automatic system. The subsections below describe each type of feature group, together with its individual features. Additional details can be found in the available code.

4.1. Linguistic features (baseline)

This class of features is inspired by readability research and its application to estimating question difficulty, forming a strong baseline for comparison to other approaches (Dubay, 2004; McNamara et al., 2014; Yaneva and Evans, 2015; Yaneva et al., 2019). It includes the following subcategories.

Lexical Features, such as counts, incidence scores and ratios for *ContentWord, Noun, Verb, Adjective*, and *Adverb*; *Numeral Count; Type-Token Ratio; Average Word Length In Syllables*; and *Complex Word Count* ($>$ 3 syllables).

Syntactic Features: These were implemented using information from the Stanford NLP Parser (Manning et al., 2014): *Average Sentence Length (words); Average Depth Of Tree; Negation Count; Negation In Stem*; *Negation In the Lead-In Question*; *NP Count*; *NP Count With Embedding* (the number of noun phrases derived by counting all the noun phrases present in an item, including embedded NPs); *Average NP Length; PP and VP Count; Proportion Passive VPs; Agentless Passive Count; Average Number of Words Before Main Verb; Relative Clauses* and *Conditional Clauses Count*.

Semantic Ambiguity Features: This subcategory concerns the semantic ambiguity of word concepts according to WordNet (WN), as well as medical concepts according to the UMLS (Unified Medical Language System) Metathesaurus (Schuyler et al., 1993). The features include *Polysemic Word Index; Average Number of Senses of: Content Words, Nouns, Verbs, Adjectives, Adverbs; Average Distance To WN Root for: Nouns, Verbs, Nouns and Verbs; Total No Of UMLS Concepts; Average No Of UMLS Concepts; Average No Of Competing Concepts Per Term* (average number of UMLS concepts that each medical term can refer to).

Readability Formulae: *Flesch Reading Ease* (Flesch, 1948); *Flesch Kincaid Grade Level* (Kincaid et al., 1975); *Automated Readability Index (ARI)* (Senter and Smith, 1967); *Gunning Fog index* (Gunning, 1952); *Coleman-Liau* (Coleman, 1965); and *SMOG* index (McLaughlin, 1969).

Cognitively-motivated Features: These are calculated based on information from the MRC Psycholinguistic Database (Coltheart, 1981), which contains cognitive measures based on human ratings for a total of 98,538 words. These features include *Imagability*, indicating the ease to construct a mental image of that word; *Familiarity*, or how familiar the word seems to an adult; *Concreteness*; *Age Of Acquisition*; and finally *Meaningfulness Ratio Colorado* and *Meaningfulness Ratio Paivio*. The meaningfulness rating assigned to a word indicates how associated the word is to other words.

Word Frequency Features: These include *Average Word Frequency*, as well as threshold frequencies such as words not included in the most frequent words on the BNC frequency list (*Not In First 2000/ 3000/ 4000 or 5000 Count*).

Text Cohesion Features: These include counts of *All Connectives*, as well as *Additive; Temporal;* and *Causal* Connectives, and *Referential Pronoun Count*.

4.2. Embeddings

We experiment with two types of embeddings: Word2Vec (300 dimensions) (Mikolov et al., 2013) and ELMo (1,024 dimensions) (Peters et al., 2018). The results presented in this paper refer to embeddings generated using approximately 22,000,000 MEDLINE abstracts,which were found to outperform other versions of the embeddings extracted from generic corpora (Google News Corpus[4] for Word2Vec and 1B Word (Chelba et al., 2013) for ELMo). The embeddings were aggregated at item level using mean pooling, where an average item embedding is generated from the embeddings of all words.

[4]https://news.google.com

4.3. Information Retrieval (IR) features

This group of features measures how difficult it is for an automatic question-answering (QA) system to answer the items correctly. The hypothesis behind the design of these features is that MCQs which are more difficult to answer automatically would also be more difficult for humans (Ha and Yaneva, 2019). This idea stems from observations such as the fact that both humans and machines need to retrieve a given information in order to answer the questions, where humans use their subject knowledge, while machines query a database. Another parallel between the two processes is the need to reason over facts, where humans still have a great advantage. In spite of the limitations of current QA systems, it is conceivable that since humans and machines need to perform similar processes in order to answer a question, these processes might be challenged by the same types of questions. Even if such parallels are not found, testing this working hypothesis is a useful by-product of the current research, as it may inform better strategies for approaching automatic QA tasks. To test the hypothesis, we develop and train a full automatic question-answering system following approaches presented in Clark et al. (2018). After that, we use the scores obtained by the system to extract the following features, as explained below.

First, we index the abstracts of medical articles contained in the MEDLINE[5] database using Lucene[6] with its default options. Then we query the database as follows. For each test item we build several queries, where each query contains the stem and one answer option. We use three different settings: i) All words, ii) Nouns only, and iii) Nouns, Verbs, and Adjectives (NVA). We then get the top 5 MEDLINE documents returned by Lucene as a result of each query and calculate the sum of the retrieval scores. These scores represent the content of the IR features (*Stem Only, Stem + Correct Answer*, and *Stem + Options*, where for each of these configurations we have a different feature for All words, Nouns only and NVA.). The scores reflect how difficult it is for a QA system to choose the correct answer. Specifically, if the IR scores of *Stem + Correct Answer* are much higher than those of *Stem + Options*, then it is easy for the QA system to answer that item correctly by picking the option that has the highest scores. This information can thus be used to predict item difficulty.

5. Experiments

We explore and compare two approaches to predicting item survival, namely: i) modeling P-value and Rb individually and applying a function to select which items to retain, and ii) modeling survival as a binary classification between surviving and non-surviving items. Both these approaches are compared to three baseline models.

5.1. Baselines

The results of various models are compared to the following baselines.

- **Random:** Since the surviving and non-surviving classes are of the same size, the random baseline is F1 = 0.5. For predicting P-value and Rb individually, the model performance is first compared to the output of the ZeroR rule (assigning the mean of the distribution as a predicted value to each instance), the cut-off function is applied to classify the items based on the predicted values and then classification accuracy is measured and compared to the random baseline.

- **Item length in words:** This baseline is applied to rule out the possibility that item survival is simply a function of item length (e.g., that longer items may be more difficult). Using item length in words as a single variable in an Random Forests model (algorithm chosen for comparability, see below) in a 10-fold cross validation set up results in F1 = 0.51.

- **Linguistic features:** This rich baseline consists of the full set of linguistic features as input to a Random Forests classifier (see below) (10-fold CV) and the result is an F1 score of 0.54. This result significantly outperforms both the random baseline ($p < 0.0001$) and the item length one ($p < 0.001$), making the Linguistic feature model the strongest of our baselines.

	P-value		Rb	
	r	RMSE	r	RMSE
NN 3 dense layers	0.23	27.8	0.15	19.83
SVM (regr.)	0.22	33.13	0.08	24.31
Gaussian Processes	0.22	29.71	0.11	19.98
Linear regression	0.18	36.86	0.05	25.83
Random Forests	**0.31**	**26.82**	**0.13**	**18.19**

Table 2: Evaluation of five algorithms (among others) for predicting P-value and Rb (NN parameters include: 3 dense layers of size 100, activation function: RELU, loss function: MSE, weight initialization Xavier and learning rate = 0.001. Trained for 500 epochs with early stopping after 10 epochs with no improvement.)

5.2. Algorithm selection

First, we assign the items to a training (60%), validation (20%), and test (20%) sets. We then evaluate various neural and non-neural models on the validation set. As shown in Table 2, the neural approaches consistently performed better than most non-neural ones in predicting P and Rb, but were convincingly outperformed by the Random Forests (RF) algorithm (Breiman, 2001). For example, the RF algorithm outperformed a neural network with three dense layers by reducing RMSE for P-value with approximately one point (Table 2). For the classification task, non-neural approaches proved consistently more suitable than neural ones, with RF again peforming best (F1 = 55.8). The rest of the results on algorithm selection for the classification task are presented in Table 3. Subsequent experiments involved testing various feature combinations.

5.3. Modeling P an Rb

This subsection presents results for item survival obtained by modeling P and Rb individually. Of all feature combinations, best performance was achieved using all features. P is predicted with r = 0.31 for the validation and 0.26 for the

[5]https://www.nlm.nih.gov/bsd/medline.html

[6]https://lucene.apache.org/

	F1
NN with 3 dense layers	47.2
LSTM (3 layers)	47.5
BayesNet	55.1
SVM	54.3
RF	**55.8**

Table 3: Evaluation of five algorithms (among others) for predicting item survival in a classification task (NN parameters include: 3 dense or LSTM layers of size 100, activation function: RELU, weight initialization Xavier and learning rate = 0.001. Trained for 500 epochs with early stopping after 10 epochs with no improvement.)

	P-value RMSE		Rb RMSE	
	Valid.	Test	Valid.	Test
ZeroR	28.18	29.1	17.48	18.47
RF	26.82	28.08	17.28	18.21
P (sign).	< 0.0001	< 0.001	0.341	0.027
CI	-1.7; -.68	-.1, -.03	-.36, .13	-.43, -.02

Table 4: Root Mean Squared Error (RMSE) for P-value and Rb (full feature set) with bootstrapped significance

test sets, while the correlation is much lower for Rb (0.16 and 0.17). As shown in Table 4, the reduction in RMSE represents a significant improvement over the ZeroR baseline for P, but not for Rb. While some of these improvements are statistically significant, their practical significance for predicting survival is low. Based on the predicted values, almost all items would be classified as surviving (F1 = 50). As a next step, we model item survival as a classification task.

5.4. Classification

Approaching survival as a classification task, we use the RF algorithm with various feature combinations as input. The evaluation is based on 10-fold cross-validation. The results for various combinations are presented in Table 6.

As can be seen from the table, all feature combinations led to a statistically significant improvement over both the Random baseline and the Item length baseline, with the exception of the IR features, which outperform only the Random baseline ($p = 0.0047$, 95% CI: 0.79, 4.39). The strongest baseline, the Linguistic feature model (F1 = 54), is outperformed by the full feature set with an F1 of 55.8 ($p = 0.04$, 95% CI: 0.0072, 3.5912). The Area Under ROC Curve for this result is 57.8. Notably, ELMo achieved a result that was very close to the best result (55.6) but it did not outperform the Linguistic feature baseline. Nevertheless, ELMo emerged as the strongest predictor among all features. The fact that the best result is achieved using all feature types indicates that they complement rather than overlap each other.

5.5. Error analysis

Analysis of the errors for the full feature set reveals that the model results in more false positives than false negatives, where 40% of the false positives occur due to failure to recognize negative Rb values as non-surviving. The set of false negatives indicates that items with higher P-values tend to be mistakenly classified as non-surviving (μ = .73, min = .31, max = .95). Roughly 45% of the items with negative Rb values are mistakenly classified as surviving and these are mostly instances with P-values within the surviving range.

To understand the behaviour of the different feature classes better, we analyse the model output of each feature class (Linguistic, IR, Word2Vec and ELMo). The linguistic features perform more or less comparable to the other types in terms of identifying true positives but have the lowest rate of true negatives, especially in predicting the Rb component. Conversely, ELMo do best among all feature types in recognizing that negative Rb values should be labeled as non-surviving but has the highest rate of false positives above the .95 threshold. No specific pattern emerged for the IR features and Word2Vec.

	F1
Random	50
Item length	51
Linguistic	54

Table 5: Baselines

	F1
IR	52.6
Ling + W2V + ELMo	54.5*
W2V	54.6*
W2V + ELMo	54.8*
IR + W2V	55.2*
Ling + ELMo	55.2*
IR + Ling + W2V	55.3*
IR + ELMo	55.4*
IR + W2V + ELMo	55.4*
IR + Ling + ELMo	55.5*
ELMo	55.6*
All	**55.8****

Table 6: Ranking of the results from various feature combinations for classification experiments. All combinations outperform the random baseline with a statistically significant difference. Values marked with * signify statistically significant improvement over the Item length baseline (F1 = 51) and ** signifies a statistically significant improvement over the Linguistic features baseline (F1 = 54).

5.6. Feature importance

As can be seen from Table 6, ELMo performs best among all feature types, followed by Word2Vec, the linguistic features and the IR features at the last place. A more detailed analysis of the linguistic and IR features is presented in Table 7 which shows the top five highest correlated features from each set.

Individual features are very weakly correlated with the class labels, indicating that no single feature has the predictive power to dominate the models. This result was corroborated with a backward feature elimination study for the IR set, where the removal of individual IR features did not lead to notable improvement in the performance of the set but instead reduced the accuracy.

Top IR features	r	Top Linguistic features	r
Stem NVA max	.041	Content Word inc.	.078
Stem Noun max	.04	Noun inc.	.067
Stem Noun sum	.04	Av. Sense Content W.	.057
Stem NVA sum	.039	NP inc	.056
St. + Correct NVA	.038	Av. NP length	.056

Table 7: Top 5 features with highest correlation with class labels (NVA = nouns, verbs and adjectives, sum = sum of similarities, max = maximum similarity, inc = incidence)

6. Discussion

The classification models achieved modest, but statistically significant better results compared to three baselines: random assignment, item length and a strong baseline of 113 linguistic features. The fact that adding the IR features and embeddings achieved significant improvement over the linguistic features is an encouraging evidence that the models captured difficulty beyond the linguistic complexity represented by this baseline. In terms of task definition, the results from the classification approach were better than those from modeling P-value and Rb individually, suggesting that there may be value in learning the two parameters simultaneously.

While the IR features alone performed worse than the linguistic ones, there are two interesting observations that can be made regarding their role. First, they significantly outperformed the random assignment baseline, which shows that they did provide some signal. Second, they were included in the best performing model, as well as the majority of the better models, which shows that the signal they provided did not overlap with that of the embeddings or linguistic features. This is an evidence that there is value in approaches to question difficulty estimation that rely on input from automatic question-answering systems and that there are some similarities between questions that are difficult for machines and those that are difficult for humans. The exact extent to which this is the case is the subject of a separate investigation.

The most challenging part for the models was the correct classification of items with a negative Rb component which had P-values within the range of surviving items. This is explained by the ability of the models to predict the P-value with higher accuracy than the Rb component, as shown by modeling the two parameters individually. As revealed by the error analysis, the great advantage of ELMo compared to other feature types is the comparatively higher accuracy at predicting Rb. Nevertheless, ELMo did worse at identifying non-surviving items at the high end of the P-value spectrum. It is these differences in the predictive power of the different feature sets that make them complement each other rather than overlap. In terms of feature importance, ELMo performed best, followed by Word2Vec, the linguistic features and the IR features. As shown in the feature analysis section, no individual features from the linguistic or IR sets had a significantly higher correlation with class labels than the rest.

One of the more serious limitations of this study is the relatively low overall accuracy achieved by the models compared to other classification tasks. This is largely explained by the difficulty of the problem, since predicting item survival for highly-specialised questions with no varying reading levels requires going beyond the measurement of linguistic complexity. While this limitation remains true, predicting item survival represents low-risk and high-benefit strategy, as even small improvements in accuracy can lead to saving substantial resources from pretesting. This can be achieved if items are first filtered automatically and pretesting slots are then assigned to items that are more likely to pass the thresholds. For example, the best result of 55.8 translates to having 3,304 correctly classified items, which is 345 more than the random assignment baseline. In practice, one standard nine-hour exam can only afford to have one block of pretesting items, which is 40 items (the bottleneck problem referred to in the introduction section) and each item is seen by an average of 328 examinees. This illustrates the suitability of NLP to help with this practical task by reducing its cost without introducing any major risks, especially in the context of evaluating automatically generated questions.

One of the strengths of the proposed approach is its generic nature, which means that it could be applicable to other exams. While such data sets are typically not freely available because of test security (exam questions become unusable if there is a chance that examinees might have seen them in advance), they exist in most high-stakes exams that test subject knowledge using the multiple-choice format. It is therefore conceivable that the proposed approach could be used in other exams.

These results are a first step towards the challenging but highly beneficial application of predicting item survival. Future work includes the exploration of multi-task learning for predicting P and Rb simultaneously, as well as searching for better predictors for the Rb component.

7. Conclusion

This paper presented a new practical application for NLP, namely predicting item survival from item text. The task involves differentiating between good quality exam questions and ones that are too easy or too difficult, as well as ones that do not improve the quality of the overall test score. Four feature types were extracted, consisting of 113 linguistic features (baseline), Word2Vec and ELMo embeddings, as well as features related to the difficulty a QA system would have with answering the questions. Results indicated statistically significant improvement over the linguistic model, a random assignment baseline and an item length baseline, by using all feature types in a binary classification task. ELMo had the highest predictive power, followed by Word2Vec, the linguistic features and the IR features. The proposed approach is generic and has the potential to generalize over other high-stakes exams that test subject knowledge through multilple-choice questions.

8. Bibliographical References

Alsubait, T., Parsia, B., and Sattler, U. (2013). A similarity-based theory of controlling mcq difficulty. In *e-Learning and e-Technologies in Education (ICEEE), 2013 Second International Conference on*, pages 283–288. IEEE.

Baldwin, P., Yaneva, V., Mee, J., Clauser, B. E., and Ha, L. A. (2020). Using natural language processing to predict item response times and improve test construction. *Journal of Educational Measurement*.

Beinborn, L., Zesch, T., and Gurevych, I. (2015). Candidate evaluation strategies for improved difficulty prediction of language tests. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–11.

Bloom, B. S. et al. (1956). Taxonomy of educational objectives. vol. 1: Cognitive domain. *New York: McKay*, pages 20–24.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brennan, R. L. (2006). *Educational Measurement. ACE/Praeger Series on Higher Education.* ERIC.

Chelba, C., Mikolov, T., Schuster, M., Ge, Q., Brants, T., Koehn, P., and Robinson, T. (2013). One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. (2018). Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Coleman, E. B., (1965). *On understanding prose: some determiners of its complexity*. National Science Foundation, Washington, D.C.

Coltheart, M. (1981). The mrc psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A*, 33(4):497–505.

Dubay, W. H. (2004). *The Principles of Readability*. Impact Information.

Flesch, R. (1948). A New Readability Yardstick. *Journal of Applied Psychology*, 32(3):221–233.

Gierl, M., Lai, H., and Zhang, X. (2018). Automatic item generation. In *Encyclopedia of Information Science and Technology, Fourth Edition*, pages 2369–2379. IGI Global.

Gunning, R. (1952). *The technique of clear writing*. McGraw-Hill, New York.

Ha, L. A. and Yaneva, V. (2018). Automatic distractor suggestion for multiple-choice tests using concept embeddings and information retrieval. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 389–398.

Ha, L. A. and Yaneva, V. (2019). Automatic question answering for medical mcqs: Can it go further than information retrieval? RANLP.

Ha, L., Yaneva, V., Balwin, P., and Mee, J. (2019). Predicting the difficulty of multiple choice questions in a high-stakes medical exam. Association for Computational Linguistics.

Huang, Z., Liu, Q., Chen, E., Zhao, H., Gao, M., Wei, S., Su, Y., and Hu, G. (2017). Question difficulty prediction for reading problems in standard tests. In *AAAI*, pages 1352–1359.

Kincaid, J. P., Fishburne, R. P., Rogers, R. L., and Chissom, B. S. (1975). *Derivation of new readability formulas (Automatic Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy enlisted personnel*. CNTECHTRA, 8-75 edition.

Loukina, A., Yoon, S.-Y., Sakano, J., Wei, Y., and Sheehan, K. (2016). Textual complexity as a predictor of difficulty of listening items in language proficiency tests. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3245–3253.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

McLaughlin, H. G. (1969). SMOG grading - a new readability formula. *Journal of Reading*, pages 639–646, May.

McNamara, D. S., Graesser, A. C., McCarthy, P. M., and Cai, Z. (2014). *Automated evaluation of text and discourse with Coh-Metrix*. Cambridge University Press.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Nadeem, F. and Ostendorf, M. (2017). Language based mapping of science assessment items to skills. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 319–326.

Padó, U. (2017). Question difficulty–how to estimate without norming, how to use for automated grading. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–10.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Schuyler, P. L., Hole, W. T., Tuttle, M. S., and Sherertz, D. D. (1993). The umls metathesaurus: representing different views of biomedical concepts. *Bulletin of the Medical Library Association*, 81(2):217.

Senter, R. J. and Smith, E. A. (1967). Automated Readability Index. Technical Report AMRL-TR-6620, Wright-Patterson Air Force Base.

Yaneva, V. and Evans, R. (2015). Six good predictors of autistic text comprehension. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 697–706.

Yaneva, V., Baldwin, P., Mee, J., et al. (2019). Predicting the difficulty of multiple choice questions in a high-stakes medical exam. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–20.