8th ISCA Workshop on Speech and Language Technology in Education (SLaTE 2019)
20-21 September 2019, Graz, Austria

Using K-Means in SVR-Based Text Difficulty Estimation

Ray Budd[1], *Tamas Marius*[2], *Paul Gatewood*[1], *Doug Jones*[1]

[1]MIT Lincoln Laboratory [2]Defense Language Institute Foreign Language Center

{Raymond.Budd,Paul.Gatewood,daj}@ll.mit.edu, tamas.marius@dliflc.edu

Abstract

A challenge for second language learners, educators, and test creators is the identification of authentic materials at the right level of difficulty. In this work, we present an approach to automatically measure text difficulty, integrated into Auto-ILR, a web-based system that helps find text material at the right level for learners in 18 languages. The Auto-ILR subscription service scans web feeds, extracts article content, evaluates the difficulty, and notifies users of documents that match their skill level. Difficulty is measured on the standard ILR scale with language-specific support vector machine regression (SVR) models built from vectors incorporating length features, term frequencies, relative entropy, and K-means clustering. [1]

1. Introduction

Text difficulty and readability have been studied for many years, and various approaches have been developed to measure difficulty ranging from traditional methods that leverage shallow length features to more advanced natural language processing and machine learning techniques. Throughout this long history, the focus has primarily been on methods applicable to one or a few specific languages, first language (L1) acquisition, and sentence simplification. There has been less work focused on second language (L2) learning, approaches that can be generalized to a wide variety of languages, and the development of end-to-end systems to support self-directed study and authentic material collection. There has also been little focus on the development of corpora in many languages that use a common system to designate text difficulty on a scale ranging from beginner to near-native ability.

1.1. The ILR Scale

The Interagency Language Roundtable Scale is a broadly used standard that can be leveraged for L2 learning. The U.S. Government established the Interagency Language Roundtable (ILR) to support foreign-language related activities. The ILR maintains the ILR Scale [1] which consists of 5 increasing proficiency levels: the Level 1 learner understands isolated words and phrases; Level 2 has a basic working knowledge of the language; Level 3 can satisfy basic professional needs; Level 4 can handle more advanced professional needs; Level 5 is nearly equivalent to an educated native speaker. The ILR Scale also establishes intermediate 'plus' levels (1+, 2+, etc). These indicate that proficiency exceeds a given level, but does not meet all requirements for the next level.

The ILR scale has descriptions at each level that outline the semantic and grammatical characteristics exhibited by text at the level. It was developed to measure foreign language proficiency of adult second language learners from basic to very advanced. ILR text leveling is frequently performed by expert linguists who are native speakers of the target language and trained in ILR rating. Similar to Shen et al. [2], we adopt the ILR scale as the framework to measure text complexity.

1.2. Extending the Auto-ILR System

In this paper, we build on previous work introduced by Shen et al. [2] to present an integrated system, Auto-ILR, capable of assessing text complexity in 18 languages and selecting appropriately challenging reading material for a L2 learner, educator, or test creator. The previous system applied machine learning using supervised methods with minimal reliance on language-specific characteristics, and supported four languages: English, Dari, Arabic, and Pashto. Our contributions are: (1) the expansion of the supported languages from four to 18; (2) the adaptation of the feature vector representing a document in the SVR model to include additional features like the K-means cluster of the document; (3) the integration into a production system that has evaluated tens of thousands of web articles, and is used by thousands of students; (4) the expansion of the corpora of approximately 5,500 truth marked texts in four languages described by Shen et al. [2] into a set spanning 18 languages and including over 31,000 texts.

2. Background and Related Work

Kevyn Collins-Thompson provided an extensive survey of historical and recent trends in text readability in 2014 [3]. The survey makes two observations particularly relevant to our goals: (1) there are no large, high-quality corpora available, especially covering many levels for non-English languages; (2) L2 learners are different than L1 learners. The survey also identifies a lot of activity in non-English languages, but the work is usually focused on L1 learners and supports one or a few languages using language-specific techniques.

Text readability has a long history spanning 70 years from traditional measures like Flesch-Kincaid to modern approaches that use machine learning. Traditional measures typically incorporate shallow text features, like average word and sentence length. For example, Flesch-Kincaid uses average word count per sentence, and syllables per word to provide an expected US grade [4].

Shortly after 2000, efforts shifted toward applying machine learning algorithms to evaluate readability. Schwarm and Ostendorf's early experiments in 2005 used support vector machines to classify English articles on a scale corresponding to US school grades 2 through 5 [5]. Their system significantly outperformed the Flesh-Kincaid formula. However, it was limited to English and a subset of L1 learners. Peterson and Ostendorf later expanded this work and explored the applicability of regression models as an alternative to classification [6].

[1]This work is sponsored by the Defense Language Institute Foreign Language Center under Air Force Contract FA8702-15-D-0001. Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

10.21437/SLaTE.2019-16

In 2013, Shen et al. examined the prediction of readability for L2 learners with a focus on language independence [2]. This work is followed by Shen and Salesky's work [7] and serves as a foundation for our activities. They examine ILR annotated text in four languages (English, Dari, Arabic, and Pashto), and compare two regression algorithms: Margin-Infused Relaxed Algorithm (MIRA) and linear support vector machine regression (SVR). Shen and Salesky's later paper [7] extended the feature vector to include relative entropy, word, and class-based language models. These prior systems support fewer languages, and have unneeded complexity that we discuss in 4.

In 2014, Roukos et al. also explored language independent evaluation of readability [8]. Their approach also used the ILR scale, but leveraged Machine Translation and derived features from the translated text. They experimented with a corpus of 4,500 documents in 54 non-English languages with manual English translations and ILR annotation. Their work introduces an interesting technique, but is limited to languages with Machine Translation, and has limited training/validation materials. With 4,500 documents split across 54 languages, less than 84 documents would be available per language on average.

In 2018, Jiang et al. explored the application of word embeddings to readability assessment in English and Chinese [9]. Word embeddings capture additional contextual information not available with the traditional bag-of-words models that are used in many recent systems. They evaluated their system on 1,400 documents containing mainly textbook materials with the Chinese augmented by primary school student essays. This is a novel application of word embeddings, but is limited to Chinese, and does not leverage authentic materials.

3. System Architecture

The Auto-ILR system comprises three web-based components: a user-facing subscription service, a web feed reader, and a text assessor. Learners, and instructors interact with the system via the subscription service. With this service, a learner registers to receive articles of a desired difficulty level from web feeds of interest. The feed reader monitors web feeds (e.g. ATOM or RSS), retrieves new articles, and extracts the main text of the article leveraging the Boilerpipe library. This library is based on techniques introduced by Kohlschütter, et al. [10] which use classifiers trained mainly on shallow text features like average sentence and word length to identify and discard boilerplate text from web pages. After text extraction, the feed reader uses the difficulty assessor to identify the text's difficulty, and provides the learner with links to articles matching his or her preferences.

4. Text Assessor

Auto-ILR's text assessor performs difficulty estimation according to the ILR scale, and treats it as a continuous, linear scale. Values range from 1 to 4 with plus levels considered 0.5 from the base (e.g. 1+ becomes 1.5). The assessor uses linear support vector machine regression (SVR) models trained on truth-marked data in each language, and generates predictions on a continuous scale that are mapped to ILR levels. It's interesting to note that the systems in [2] and [7] use MIRA, but we prefer SVR. MIRA allows for online learning, but SVR has comparable performance and reduces complexity. We also find that the small data sets allow frequent retraining.

The text assessor is implemented in Python, and uses Scikit-learn [11] as the framework for the training pipeline. We use its internal SVR based on LIBSVM [12] along with other features. The assessor is wrapped as a web service that takes individual articles and returns the discrete ILR level along with the regressed value originally predicted.

4.1. Training and Performance Metric

Our training pipeline consists of basic tokenization and data cleansing, extraction of language agnostic features, and training of the SVR model. The tokenization and cleansing process consists of splitting words on white space and sentences on punctuation. For word tokens, we convert to lowercase, remove punctuation, and discard tokens that contain numbers or do not start with the target language script. For language orthographies that do not have white space word separation, (e.g. Chinese), a dictionary-based tokenization scheme is used.

To train the model for a language, we divide the documents into train and test sets using a 75-25 split. We use stratified 5-fold cross validation on the training set with final evaluation against the test set. We measure performance with Mean Squared Error (MSE) similar to Shen et al. [2]. During hyperparameter tuning, we explore different values for the SVR penalty (C), epsilon margin (ϵ), and minimum and maximum term frequency in the word vectors.

4.2. Text Features

Features extracted from text include Z-score normalized length features, relative entropy, word frequency, and the K-means cluster of each document. Length features are average word and sentence length. Relative entropy, also known as the Kullback-Leibler divergence, is a measure of how different two probability distributions are. In our application, the first probability distribution is a unigram language model for a document, i.e. a distribution over the terms in the document where the probability is based on the number of times the term is found and the total number of document terms. The second is a uniform probability distribution, i.e. a probability distribution over the terms in the document that assumes each term is equally likely to appear. We measure word frequency with TF-IDF (Term Frequency - Inverse Document Frequency) bag-of-words vectors where TF is logarithmically scaled to reduce the impact of high frequency terms (LOG-TF).

K-means clustering is done using Lloyd's algorithm [13], and partitions the training documents into K clusters based on word frequency. The algorithm starts by selecting an initial set of K-means (cluster centroids), then repeatedly: (1) Assigns each document to the closest cluster; then (2) Calculates the new cluster mean (centroid). K-means uses Euclidean distance to identify the closest cluster, which is not accurate in the high dimensional space of a TF-IDF vector. To improve accuracy, we perform an initial feature reduction via Principal Component Analysis (PCA) to reduce the dimensionality. PCA accomplishes this by combining features in the high dimensional space into a smaller number of principal components, features which represent the maximum variance of the combined features.

While K-means is an unsupervised algorithm, we can use document truth markings to better understand the algorithm's performance on our documents. Figure 1 shows a Voronoi diagram of 7 K-means clusters created for 1,385 English documents truth marked with 7 ILR levels. Each point is a document, where the X and Y axis are the 1st and 2nd principal components calculated by PCA from the document TF-IDF vectors. While there is some overlap, you can see general groupings: level 1 and 1+ appear mainly in clusters A and C, level 2 in D, with higher level documents in B and F.

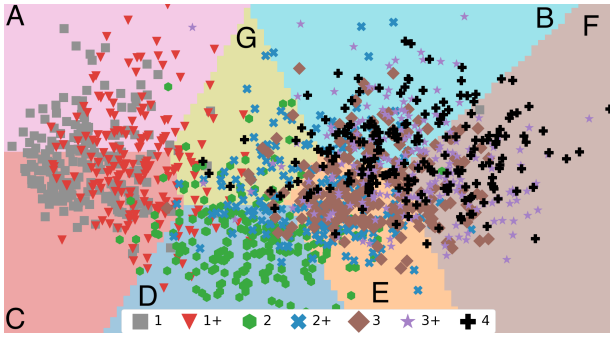

Figure 1: *Voronoi diagram for the English corpus. X and Y values are the 2 principal components identified for each document and plotted against the 7 K-means clusters (A-G). Each series represents a different document difficulty (ILR) level.*

The results for 1,420 Spanish documents shown in Figure 2 are less clear, but still informative. There is a trend to put harder documents in clusters A and C, and easier documents in E and F. Mid-level documents 2, 2+ and 3 are scattered in the center.

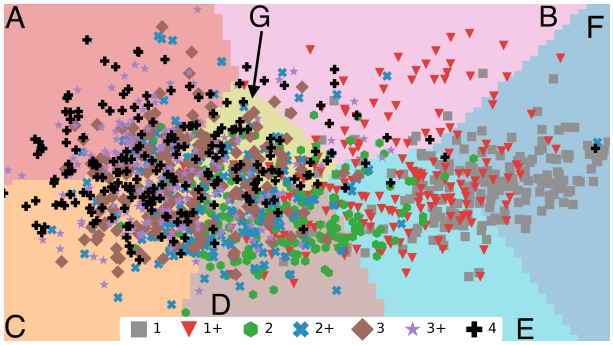

Figure 2: *Voronoi diagram for the Spanish corpus. X and Y values are the 2 principal components identified for each document and plotted against the 7 K-means clusters (A-G). Each series represents a different document difficulty (ILR) level.*

The results can be further quantified by measuring the homogeneity, completeness, and V-measure (i.e. the harmonic mean of the two) [14]. The three metrics range from 0.0 to 1.0 with higher values considered better. Homogeneity is a measure of cluster uniformity where 1 indicates that each cluster contains members of only one class. Completeness is a measure of how many members of each class are represented by the same cluster where 1 means that each class falls entirely within a cluster.

See Table 1 for the K-means metrics calculated on 5 Auto-ILR language data sets. It's interesting to note the homogeneity is low in general, meaning documents in different classes appear in the same cluster. This implies that the resulting K-means model will not be an effective feature because the cluster value does not discriminate between different ILR levels.

We explored cluster sizes in various increments ranging from 7 to 1000, and found 300 to be the upper threshold that shows performance gains. See Table 2 for the K-means metrics calculated when generating 300 clusters. The result is a boost to homogeneity at the cost of completeness in each cluster, but this is desirable for us. It means more clusters, but each cluster more closely represents a single ILR level. It's also interesting

Table 1: *Cluster homogeneity, completeness, and V-measure for K-means with 7 clusters for 5 Auto-ILR language data sets.*

	Homogeneity	Completeness	V-Measure
English	0.368	0.499	0.424
Serbian	0.359	0.483	0.412
Arabic	0.263	0.393	0.315
Spanish	0.251	0.283	0.266
Tagalog	0.250	0.282	0.265

Table 2: *Cluster homogeneity, completeness, and V-measure for K-means with 300 clusters for 5 Auto-ILR languages data sets.*

	Homogeneity	Completeness	V-Measure
English	0.535	0.204	0.295
Serbian	0.500	0.262	0.344
Arabic	0.423	0.201	0.273
Spanish	0.388	0.324	0.353
Tagalog	0.356	0.269	0.306

to look more closely at the differences in homogeneity between languages. Of the languages shown, Serbian, English, and Arabic are the best performing; Spanish and Tagalog are the worst.

In summary, the full feature vector used in training and prediction to describe a document is shown in in Figure 3.

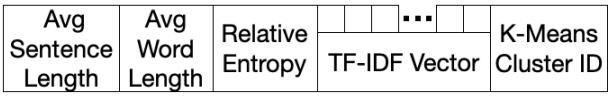

Figure 3: *The complete feature vector. Relative entropy, word and sentence lengths are Z-score normalized; the term frequency is logarithmically scaled in the TF-IDF vector.*

4.3. Data Corpora

All training and evaluation data sets are comprised of materials collected by foreign language instructors at the Defense Language Institute Foreign Language Center. To generate each data set, native speaker instructors, skilled in both the target language and the ILR scale, collected representative texts, and annotated each with an ILR level and topic. Each document received one rating. The system leverages the Arabic, Dari, English, and Pashto data sets described in Shen et al. [2], along with data sets in an additional 14 languages containing 1149 to 3652 documents each as shown in the DOCS column of Table 6.

The primary purpose of Auto-ILR is to give foreign language instructors an easy way to find texts at the appropriate level of difficulty for their students. In an early, informal comparison among seven raters for a small set of Arabic documents, we found the difference between the raters to be comparable to the difference between an average human rating and the Auto-ILR system. We decided, therefore, to collect a larger number of documents with single ratings rather than a smaller set with multiple ratings. We have begun to revisit inter-annotator agreement, and Table 3 shows the results for a second annotation of the Arabic document set. The overall accuracy is about 80%, Cohen's kappa is 0.55, and the confusion matrix shows that disagreements involve adjacent levels. The "plus" levels have been combined with the next higher level in the table.

Table 3: *Confusion matrix showing ILR ratings from level 1 through 4 (3+/4) for two annotators on the Arabic data set. Plus levels are combined with the next higher level.*

Level	1	1+/2	2+/3	3+/4
1	156	41	0	0
1+/2	9	217	49	1
2+/3	0	10	186	55
3+/4	0	0	30	238

4.4. Experiments and Performance

During system development, we explored several feature combinations to better understand the impact of K-means and IDF on baseline models. Table 4 presents the performance in MSE of three feature combinations along with the baseline that uses length features, relative entropy, and LOG-TF vectors.

Table 4: *Results of experiments shown in MSE. Baseline uses sentence and word lengths, entropy, and LOG-TF. +IDF adds IDF to the baseline, and +KM adds K-means to the baseline.*

	Baseline	+IDF	+KM	All
English	0.150	0.161	0.122	0.116
Arabic	0.152	0.172	0.136	0.124

It's interesting to see that weighting term frequency by IDF results in more error than in the baseline. To understand, we can consider the impact of IDF in relation to our corpora and the regression task. IDF reduces the influence of common terms, but when predicting text difficulty, this may not always be desired. See Table 5 for a breakdown of a few common English terms by difficulty level. The most common ('and', 'the') are seen at all levels, but some ('however', 'might') are only common in some of the levels. IDF reduces the impact of this information. We would like to further analyze the slight increase in performance with both IDF and K-means, but anticipate it may be due in part to dimensionality reduction with PCA.

Table 5: *The number of documents at each difficulty level that include some common terms in the English corpus. Some terms ('the', 'and') are common throughout, while others are limited to more difficult documents.*

Term	1	1+	2	2+	3	3+	4
the	111	178	204	196	202	198	190
and	150	187	203	195	202	198	190
however	1	0	12	37	31	73	94
how	7	13	29	97	113	119	126
might	0	2	10	45	80	96	96

See Table 6 for the average Cross-Validated performance for each language across all ILR levels in MSE. All languages meet our production threshold with a average cross-validation MSE under 0.25, and show a low standard deviation. We improve performance of the 4 languages (Arabic, Dari, English, and Pashto) from the baseline by Shen et al. [2].

Croatian and Serbian report the best performance, but this may be partially due to the annotation process. Typically many annotators are involved in each language's data collection, but these languages were annotated by a single person. We plan to validate the annotations with an independent rater in the future.

Table 6: *Current System Performance: average MSE across CV folds, standard deviation, and all documents in the corpus.*

Language	CV MSE	STDEV	DOCS
Arabic	0.1224	0.0124	1392
Dari	0.2063	0.0178	1387
English	0.1176	0.0229	1385
Pashto	0.1574	0.0146	1371
Chinese - Simplified	0.1651	0.0162	1454
Chinese - Traditional	0.1283	0.0122	2009
Croatian	0.0751	0.0052	1400
French	0.1327	0.0107	1434
German	0.1911	0.0225	1400
Korean	0.2192	0.0289	2484
Persian Farsi	0.1819	0.0255	1151
Portuguese	0.1343	0.0171	1413
Russian	0.1874	0.0340	1475
Serbian	0.0772	0.0024	2789
Spanish	0.2409	0.0271	1420
Tagalog	0.2470	0.0219	1477
Turkish	0.1259	0.0078	2753
Urdu	0.1874	0.0109	3652

The next best performers are English and Arabic. These are the first languages supported and have been used in more experiments over time, which may give them more influence in our feature selection. They also have more annotation support. The worst performing languages are Spanish and Tagalog. Part of this is likely due to the poor performance by K-means as shown in Table 2 in 4.2.

5. Applications

As described above, automatic text difficulty assessment has been integrated into Auto-ILR, a Web-based subscription service used by thousands of instructors and students. It has become an important tool to help curriculum developers at the Defense Language Institute Foreign Language Center find authentic materials online at the right difficulty for the students as they progress through their courses. Over 10,000 articles were manually evaluated in 18 languages from January until mid-May 2019, and the system has over 68,000 subscriptions by over 5,000 subscribers. Since its inception, the system has automatically leveled 1.2M documents from newspapers and websites.

6. Summary and Future Work

In this paper, we presented Auto-ILR, a production system used by thousands of learners around the world to measure text readability along the ILR scale in 18 languages. We described the novel combination of length features, relative entropy, TF-IDF vectors, and K-means clustering into a regression model that results in good performance in many languages. We also identified a unique collection of corpora that covers 18 languages, and includes over 31,000 annotated documents.

Further work is being planned to explore additional features that are both language specific and independent. We plan to experiment with word embeddings to capture additional semantic detail along with new grammatical, and syntactic features. We also intend to collect annotated corpora in more languages, and further refine the current data sets. Finally, we plan to more closely examine inter-rater agreement.

7. References

[1] "Descriptions of ilr proficiency levels," *Interagency Language Roundtable*. [Online]. Available: www.govtilr.org/Skills/ILRscale1.htm

[2] W. Shen, J. Williams, T. Marius, and E. Salesky, "A language-independent approach to automatic text difficulty assessment for second-language learners," in *Proceedings of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations*. Sofia, Bulgaria: Association for Computational Linguistics, Aug. 2013, pp. 30–38. [Online]. Available: https://www.aclweb.org/anthology/W13-2904

[3] 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.

[4] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, and B. S. Chissom, "Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel," 1975.

[5] S. E. Schwarm and M. Ostendorf, "Reading level assessment using support vector machines and statistical language models," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ser. ACL '05. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 523–530. [Online]. Available: https://doi.org/10.3115/1219840.1219905

[6] S. E. Petersen and M. Ostendorf, "A machine learning approach to reading level assessment," *Computer speech & language*, vol. 23, no. 1, pp. 89–106, 2009.

[7] E. Salesky and W. Shen, "Exploiting morphological, grammatical, and semantic correlates for improved text difficulty assessment," in *Proceedings of the Ninth Workshop on Innovative Use of NLP for Building Educational Applications*, 2014, pp. 155–162.

[8] S. Roukos, J. Quin, and T. Ward, "Multi-lingual text leveling," in *Text, Speech and Dialogue*, P. Sojka, A. Horák, I. Kopeček, and K. Pala, Eds. Cham: Springer International Publishing, 2014, pp. 19–26.

[9] Z. Jiang, Q. Gu, Y. Yin, and D. Chen, "Enriching word embeddings with domain knowledge for readability assessment," in *COLING*, 2018.

[10] C. Kohlschütter, P. Fankhauser, and W. Nejdl, "Boilerplate detection using shallow text features," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 441–450.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[12] C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[13] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[14] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007.