EnJa: a Reading Level Appropriate Japanese Reading Recommender

Link to code: https://github.com/hylee719/CS229-Project

Hanna Lee, Isaac Bevers

June 2, 2021

Introduction

Non-fluent learners of a foreign language have difficulty finding reading level-appropriate texts in the language they are learning that they want to read. EnJa solves this problem for English-speaking learners of Japanese by allowing them to easily find texts at their reading level that are similar to a text they enjoy in English.

EnJa provides a basic command-line interface that allows a user to specify their approximate Japanese reading level on a scale from one (advanced) to five (beginner) and the path to a text file containing text that they want to use to find recommendations. EnJa then calculates the Jaccardian similarity between the input text and a corpus of 293 texts that were randomly selected from Aozora Bunko, an open-source database of over 15,500 Japanese-language fiction and non-fiction books (Aozora Bunko). The texts in the corpus were translated using Google Translate's API, and pre-clustered based on their similarity to each other using k-means. The input text is assigned to the cluster that contains the corpus-text that it is most similar to. Finally, a list of the names in the cluster that the input text is assigned to that are at the user's specified reading level are displayed in order of their similarity to the input text.

While EnJa is designed for English-speaking learners of Japanese, given the right data set, it could be easily adapted to arbitrary language pairs.

Related Work

There are multiple apps and databases aimed at providing reading materials for English-speaking learners of Japanese, such as Satori and NHK Japanese News, which provide annotated Japanese stories and recent news articles (respectively) designed to help English-speaking Japanese learners (Satori,)(NHK). Furthermore, reading recommendation systems are common-place in many places such as news aggregators. Cheng and Wong used machine learning to develop an adaptive recommendation system for English-language learners (2021). However, there appears to be very little work in this area. Searching Stanford Libraries for "automated reading recommendation in foreign language" and "reading recommendation in foreign language" and a number of other exploratory searches did not reveal any substantially similar projects besides Cheng and Wong's. We could not find any projects that base recommendations on similarity to an input text in a language other than the one that reading materials are being presented in.

Dataset and Features

Since we wanted EnJa to be capable of appealing to the broadest spectrum of English-speaking Japanese learners possible, we originally planned to use a sample of the Balanced Corpus of Contemporary Written Japanese (BCCWJ), which was systematically designed to contain the widest diversity of text types possible(Maekawa et al. 2014). However, accessing BCCWJ is very difficult, so we decided to use Aozora Bunko, an online collection of Japanese texts. Since our intended user is someone learning Japanese, we felt that it was important to have easy texts such as children's books, which Aozora contains, in addition to much of Japanese modern literature.

The Aozora Bunko texts were provided as zipped .txt files that were encoded in shiftJIS. We unzipped them and converted them to UTF-8. To be able to compare Aozora texts to an English input text, we needed to translate them, which we did using Google Cloud's Translate API. Since Google only provides 300 dollars of free Cloud credits, the number of texts that we translated was constrained. Accordingly, our final dataset consists of 293 texts, which were randomly chosen from Aozora Bunko.

Each text had non-alphanumeric characters removed and was converted into a word vector. Then the Jaccardian and cosine similarity between every text was calculated and stored in a file in the form of a matrix (since this was computationally intensive because of the size of theword vectors and the dataset). Finally, PCA was used to compress and visualize the dataset in two dimensions.

Methods

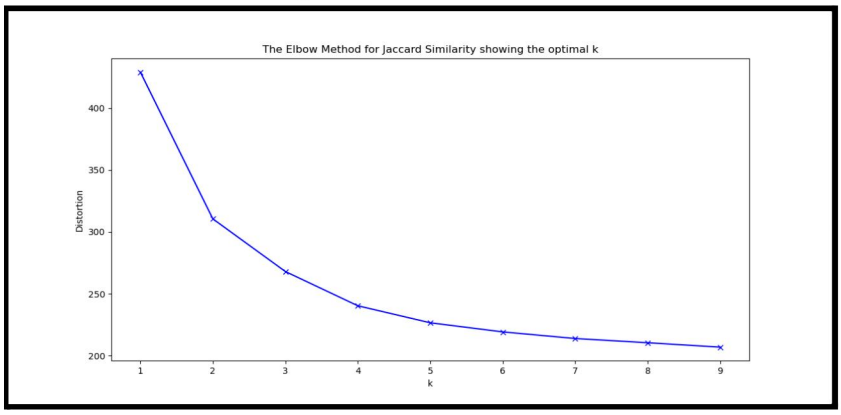
Originally, our intention was to take in an input text and output a list of same-reading-level Japanese texts in the same genre. We considered labeling each text in our dataset using genre labels available on a site such as Goodreads and using a supervised approach to predict the genre of an input text. However, this approach was problematic because it was difficult to find reliable genre labels for some texts in our dataset, and it was not possible to easily label these texts ourselves. Accordingly, we decided to use a different method for determining our genre classifications.

We found a significant amount of literature on automatic genre classification. We planned to use either a combination of logistic regression, k-nearest neighbors, and SVM; a variant of Naive Bayes; a combination of WordNet,Principal Component Analysis, and AdaBoost; or a deep learning-based approach to classify the genre of both the input text and the texts in our dataset (Panchal, 2021) (Gupta et al. 2019)(Peng et al., 2004). However, as we started tackling this approach, we encountered two problems. First, genre categories are partially language-dependent so there might not be a one-to-one mapping of Japanese text genres to English genres. Second, by using only reading level and genre as the criteria for recommendations, we would not have a way of ranking the recommended texts. While this was unlikely to be problematic considering the small size of our dataset, if our dataset was scaled up significantly, this could lead to an overwhelmingly large number of recommendations. Accordingly, we decided to find a way to rank intra-recommendation texts. Consequently, we decided to compute the word-vector similarity between the input and texts in the dataset.

We hypothesized that due to the wide diversity within genres, books within the same genre might not seem very similar to the input text when read by the user, and that word-vector similarity would be a better proxy for this. However, we also hypothesized that texts would cluster based on their word similarity to each other and that these clusters might provide a set of texts that were close enough to the input text to be interesting to the user. Accordingly, we decided to use an unsupervised approach to cluster the texts in our dataset based on their word-vector similarity to each other.

Past work using text clustering has used k-means and variants of k-means (Jing et al. 2005)f. Since k-means seems to be a standard method for text clustering and we saw no obvious merits of other, simpler models, we decided to implement it. We experimented with various numbers of centroids, which were all initialized using the scikit library "k-means++" method of optimizing centroid placement after the first one is initialized randomly, based on a probability proportional to the squared distance away from a given point's nearest existing centroid. We evaluated the performance of varying the number of centroids by seeing how often the input text was assigned to the cluster containing the five texts to which it had the highest similarity. Using the elbow method, we determined that the optimal number of centroids was five (see *Figure 1.)*, and this was consistent with our tests.

Figure 1. Elbow method results.



Our final clustering is shown in *Figure 3.*

We pre-labeled each Japanese text in our corpus with a reading level based on how many unique characters it contains. The correspondence between reading level and number of unique characters (based on JLPT reading levels plus 96 hiragana and katakana) is shown in *Table 1.* (Kanshudo).

Table1. Kanji to reading level.

N5	N4	N3	N2	N1
120 kanji	300 kanji	660 kanji	1140 kanji	2141 kanji

Experiments, Results, and Discussion

Our hypothesis about clustering in the dataset seems to be borne out to a certain degree by the increased density of some areas of *Figure 2.* However, it is difficult to be sure without more data points. We do not have a clear explanation for the crescent shape apparent in this plot. In an effort to understand to what extent our decision to use Jaccardian similarity influenced our results, we ran our analysis using cosine similarity as a point of comparison. The PCA-reduced data plot of cosine similarity for the dataset is shown in *Figure 4.* It appears that the clustering in *Figure 4.* is less obvious, although it still seems to be present, especially on the left-hand side of the plot.

Figure 2. Jaccardian similarity plot.

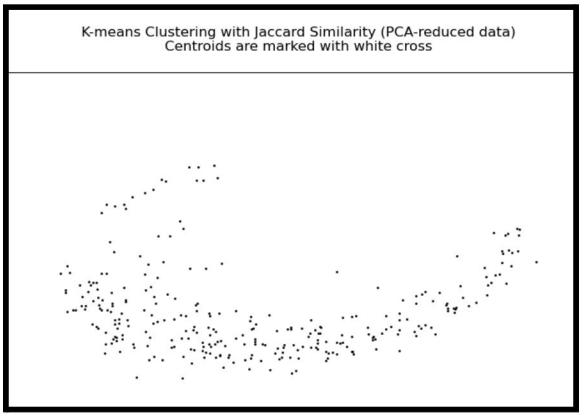


Figure 3. Jaccardian k-means clustering.

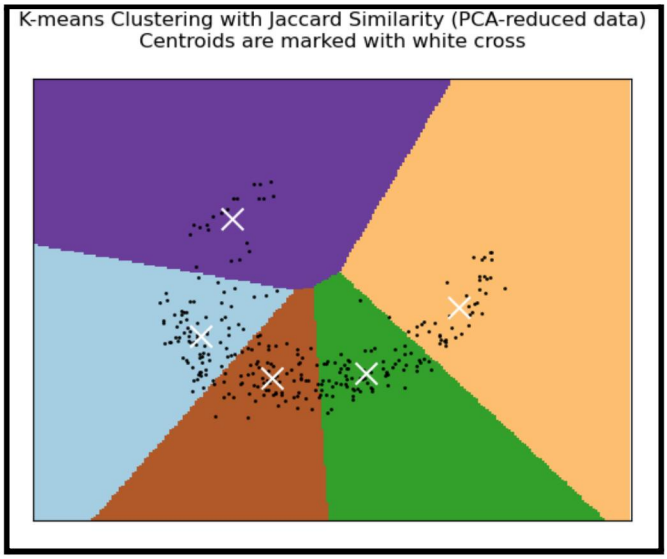
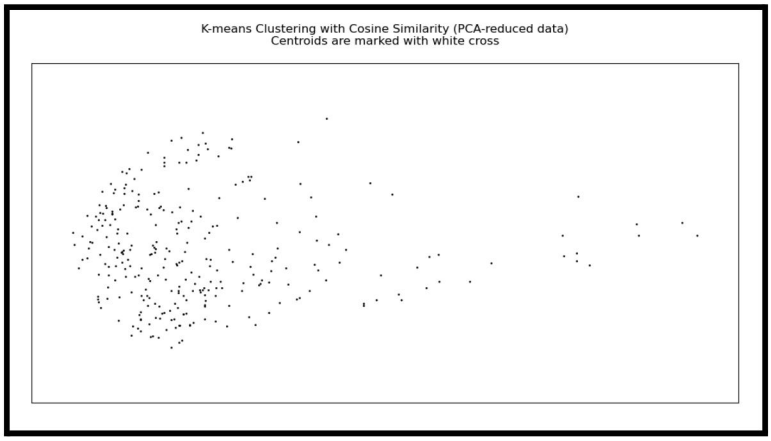


Figure 4. Cosine similarity plot.



Additionally, it seems that *Figure 4.* may be a transformation of *Figure 2.* since it could be the same shape 'smeared' out horizontally.

We also conducted k-means using cosine similarity (see *Figure 5.*). The choice of five centroids was again determined using the elbow method (see *Figure 6.*).

Figure 5. K-means clustering with cosine similarity.

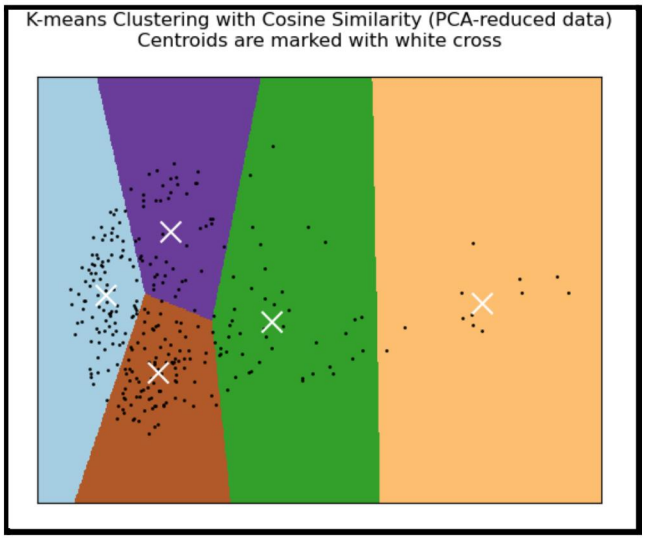
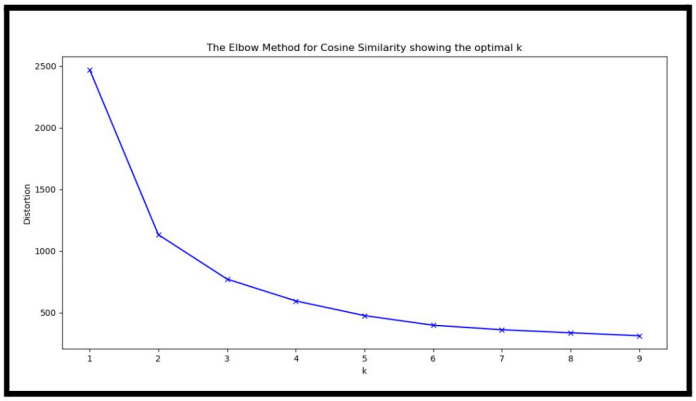


Figure 6. Elbow method for cosine similarity.



Qualitatively, it seems that the k-means clusters correspond to the density of data points better in the Jaccardian case. This was borne out by informally testing the frequency with which the text most similar to the input was present in the recommendations (i.e. it was recommended more frequently when Jaccardian similarity was used).

For some input texts, a few of the recommended texts had similarity scores over .25, which is the standard for suspecting plagiarism on turnitin.com, a popular plagiarism checking site(turnitin.com). Whether this implies that the text is qualitatively similar to the input text, however, is not entirely clear, and could only be determined by fully reading each text.

Conclusion and Future Work

The fact that clusters are not entirely distinct from each other suggests that a larger dataset would be helpful in understanding what clusters are legitimate and what are noise. A larger dataset would also increase the quality of recommendations. Additionally, the large number of data points that are not clearly associated with a cluster suggests that a probabilistic model such as a Gaussian mixture model might be more appropriate to this dataset. It is not entirely clear to what extent Jaccardian similarity is associated with a qualitative perception of similarity. Accordingly, it might be useful to conduct qualitative research on perceptions of the recommended texts.

Contributions

Hanna Lee: wrote code and utilized libraries for k-means clustering, vectorizing text documents, computing Jaccard similarity, computing cosine similarity, performing the Elbow Method to guide the number of chosen clusters, using PCA on high dimensional data to visualize k-means clustering, and retrieving a list of reading recommendations based on computed similarity scores.

Isaac Bevers: Came up with the overall project concept, collected sources, did the background research, retrieved the data-set from Aozora Bunko, converted the data, translated the data set using Google Cloud Translate API, wrote overall project code that handles the flow of the project, wrote most of the proposal, milestone, and final project report, made key modeling decisions, put together the project documents

References

Cheng, Jinyu, and Hong Wang. "Adaptive Algorithm Recommendation and Application of Learning Resources in English Fragmented Reading." Edited by Wei Wang. *Complexity* 2021 (March 30, 2021): 1–11. https://doi.org/10.1155/2021/5592534.

Google Cloud. "Cloud Translation." Accessed May 7, 2021. https://cloud.google.com/translate.

Gupta, Shikha, Mohit Agarwal, and Satbir Jain. "Automated Genre Classification of Books Using Machine Learning and Natural Language Processing." In *2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, 269–72, 2019. https://doi.org/10.1109/CONFLUENCE.2019.8776935.

Hodošček, Bor. *Borh/Aozora-Corpus-Generator*. Python, 2021. https://github.com/borh/aozora-corpus-generator.

"Interpreting the Similarity Report." Accessed June 2, 2021. https://help.turnitin.com/feedback-studio/turnitin-website/student/the-similarity-report/interpreting-the-similarity-report.htm.

Jing, Liping, Michael K. Ng, Jun Xu, and Joshua Zhexue Huang. "Subspace Clustering of Text Documents with Feature Weighting K-Means Algorithm." In *Advances in Knowledge Discovery and Data Mining*, edited by Tu Bao Ho, David Cheung, and Huan Liu, 802–12. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2005. https://doi.org/10.1007/11430919_94.

"Kanji by JLPT Level - Kanshudo." Accessed June 2, 2021. https://www.kanshudo.com/collections/jlpt_kanji.

Lee, Yong-Bae, and Sung Hyon Myaeng. "Text Genre Classification with Genre-Revealing and Subject-Revealing Features." In *Proceedings of the 25th Annual InternationalACM SIGIR Conference on Research and Development in Information Retrieval*, 145–50. SIGIR '02. New York, NY, USA: Association for Computing Machinery, 2002. https://doi.org/10.1145/564376.564403.

Maekawa, Kikuo, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. "Balanced Corpus of Contemporary Written Japanese." *Language Resources and Evaluation* 48, no. 2 (June 1, 2014): 345–71. https://doi.org/10.1007/s10579-013-9261-0.

"MeCab." In *Wikipedia*, April 26, 2021. https://en.wikipedia.org/w/index.php?title=MeCab&oldid=1019952152.

Minaee, Shervin, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. "Deep Learning Based Text Classification: A Comprehensive Review." *ArXiv:2004.03705 [Cs, Stat]*, January 4, 2021. http://arxiv.org/abs/2004.03705.

NEWS WEB EASY. "NEWS WEB EASY." Accessed June 2, 2021. https://www3.nhk.or.jp/news/easy/.

Panchal, Brijeshkumar Y. "Book Genre Categorization Using Machine Learning Algorithms (K-Nearest Neighbor, Support Vector Machine and Logistic Regression) Using Customized Dataset." SSRN Scholarly Paper. Rochester, NY: Social Science Research Network, March 16, 2021. https://papers.ssrn.com/abstract=3805945.

Peng, Fuchun, Dale Schuurmans, and Shaojun Wang. "Augmenting Naive Bayes Classifiers with Statistical Language Models." *Information Retrieval* 7, no. 3 (September 1, 2004): 317–45. https://doi.org/10.1023/B:INRT.0000011209.19643.e2.

"Satori Reader." Accessed June 2, 2021. http://www.satorireader.com.

Ueyama, Motoko, and Marco Baroni. "Automated Construction and Evaluation of Japanese Web-Based Reference Corpora," 2005.

"少納言 「現代日本語書き言葉均衡コーパス」." Accessed May 7, 2021. https://shonagon.ninjal.ac.jp/.

"青空文庫 Aozora Bunko." Accessed May 7, 2021. https://www.aozora.gr.jp/.

1 Final Project **41.5 / 50**

- **0 pts** See point adjustment / comments

- **8.5** **Point adjustment**