Finding structure in spelling and pronunciation using Latent Dirichlet Allocation

Kow Kuroda
Medical School, Kyorin University

Abstract

Latent Dirichet Allocation (LDA) was used to reveal hidden regularities in spelling and pronunciation among languages, such as English, French, German, Russian and Swahili. Words, either in spell or pronunciation, were regarded as "documents" and their characters as "terms." For spelling, spells from the languages were mixed and encoded using LDA and analyzed. For pronunciation, IPA symbols for English, French and German were encoded and analyzed in the same way. Both gave promising results.

1 Introduction

Languages differ. This is why most people have hard time in learning any of them. But at the same time, their differences are a matter of degree. Given a language, say English, some language such as French and German are *less* different from it than others such as Japanese and Russian. Languages employ similar sounds or similar spellings are clearly more alike. What this research aims at, ultimately, is to quantify such differences among languages using Latent Dirichlet Allocation (LDA) [4, 5].

2 Analysis

English spelling and pronunciation data were part of the data used in Kuroda [1] based on CMU Pronouncing Dictionary[1] and other sources. For spelling analysis, roughly 817 spells were randomly sampled from 4,279 and used in each run. For pronunciation, roughly 1,000 forms were randomly sampled from 4,199 and used in each run.

For languages other than English, spelling data were obtained from "1000 most common words" site[2] or pronunciation data from `open-dict-data`[3]. For French, 1,000 spells and 900 sounds were used. For German, 793 spells and 798 sounds were used. For Russian, 1,000 spells were used. For Swahili, 708 spells were used.

On analysis of spelling, words of English, French, German, Russian and Swahili were treated as "documents" and their orthographic characters as "terms." For terms, both character n-grams and k-skip n-grams [2] were used (k was set to the 0.8 of the longest size of the document), but n-grams are inclusive in that n-grams contain $(n-1)$-grams.

Under this, a document-term matrix (DTM) was constructed, filtered by minimum frequency = 2 and abuse threshold = 0.05, and fed to LDA with varying numbers of topics. To look for effective values, `LDA tuning`[4] was used. Then, Latent Dirichlet Allocation (LDA) [4, 5] was applied using `gensim` package (v4.2.x)[5]. Through this, all words under analysis were assigned encodings via LDA. Encodings thus obtained were visualized by t-SNE [3]. For this, `scikit-learn` package[6] was used. The reason for this choice is that grouping of items should be harder to understand with hard clustering such as hierarchical clustering than with soft clustering. For reference, results of hierarchical clustering are presented in Appendix 6.

On analysis of pronunciation, strings of IPA symbols of English, French and German were analyzed in the same way.

All data and codes used in this paper are open and available at a GitHub repository[7]: mainly, Jupyter Notebooks that implemented the analysis and the relevant data.

3 Results: Spelling

3.1 Spelling: #topics = 5

Figures 1–3 show 2D views on t-SNE 3D map (perplexity = 5) for cases for 1-, 3-gram and skippy 2-gram where

1) http://www.speech.cs.cmu.edu/cgi-bin/cmudict
2) https://1000mostcommonwords.com/
3) https://open-dict-data.github.io/
4) https://github.com/nikita-moor/ldatuning
5) https://radimrehurek.com/gensim/
6) https://scikit-learn.org/
7) https://github.com/kow-k/LDA-spell-sound-typology

#topics = 5.

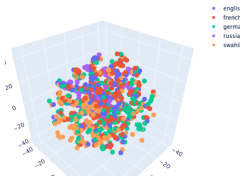

Figure 1 t-SNE 3D [#topics: 5; 1-gram; perplexity: 5]

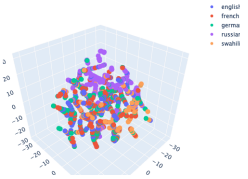

Figure 2 t-SNE 3D [#topics: 5; 3-gram; perplexity: 5]

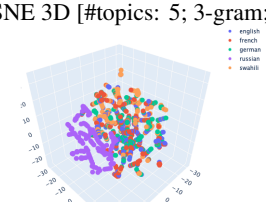

Figure 3 t-SNE 3D [#topics: 5; skippy 2-gram; perplexity: 5]

Figures 4–6 show 2D views on t-SNE 3D map (perplexity = 335) for cases where #topics = 5.

3.2 Spelling: #topics = 15

Figures 7–9 show 2D views on t-SNE 3D map (perplexity = 5) for cases for 1-, 3-gram and skippy 2-gram where #topics = 15.

Figures 10–12 show 2D views on t-SNE 3D map (perplexity = 335) for cases for 1-gram, 3-gram and skippy 2-gram where #topics = 15.

3.3 Discussion

With smaller number of topics such as 5, clustering results do not differ qualitatively for smaller perplexity values such as 5, yet they differ for larger values such as 335. It is hard to say which of Figures 1–3 is the best, but differences among the target languages seems to be successfully captured. What is common is that Russian words are isolated, which is predictable from the fact that it employs different set of characters. This is even captured in Figures 2 and 3 in which most of Russian words are somehow localized.

The localization of Swahili words is not as clear as that of Russian words, but it is observable, especially in Figure 4. What makes Figure 6 different is that it exaggerates the dissimilarity of Russian words.

With larger number of topics such as 15, results do not

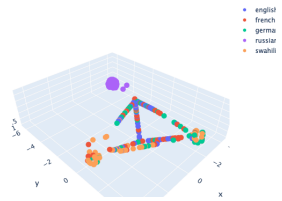

Figure 4 t-SNE 3D [#topics: 5; 1-gram; perplexity: 335]

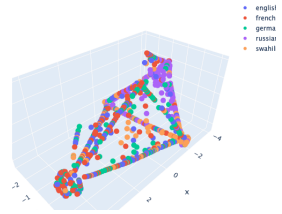

Figure 5 t-SNE 3D [#topics: 5; 3-gram; perplexity: 335]

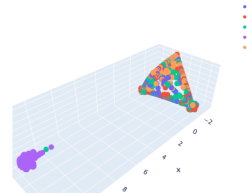

Figure 6 t-SNE 3D [#topics: 5; skippy 2-gram; perplexity: 335]

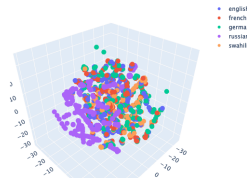

Figure 7 t-SNE 3D [#topics: 15; 1-gram; perplexity: 5]

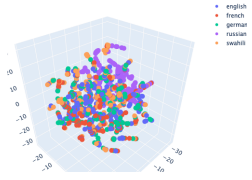

Figure 8 t-SNE 3D [#topics: 15; 3-gram; perplexity: 5]

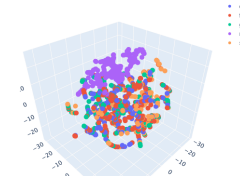

Figure 9 t-SNE 3D [#topics: 15; skippy 2-gram; perplexity: 5]

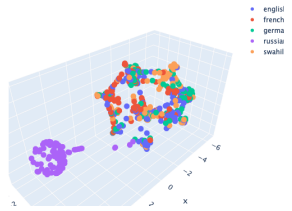

Figure 10 t-SNE 3D [#topics: 15; 1-gram; perplexity: 335]

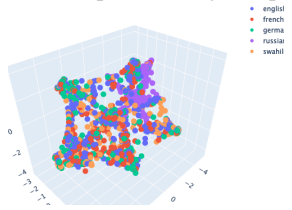

Figure 11 t-SNE 3D [#topics: 15; 3-gram; perplexity: 335]

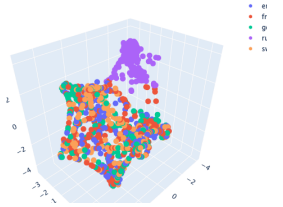

Figure 12 t-SNE 3D [#topics: 15; skippy 2-gram; perplexity: 335]

differ for smaller perplexity such as 5. This the same as the cases with #topic = 5. But the effectiveness of 1-gram-based LDA seems to be improved. Comparison among Figures 10, 11 and 12

In sum, data compression is more effective with smaller number of topics but this gives less room for representations, thereby less encoding less flexible.

4 Results 2: Pronunciation

IPA encodings of English, French and German words are analyzed using LDA.

4.1 Pronunciation: #topics = 5

Figures 13–15 show 2D views on t-SNE 3D map (perplexity = 5) for cases for 1-, 3-gram and skippy 2-gram with #topics = 5.

Figures 16–18 show 2D views on t-SNE 3D map (perplexity = 5) for cases with #topics = 335.

4.2 Pronunciation: #topics = 15

Figures 19–21 show 2D views on t-SNE 3D map (perplexity = 5) for cases for 1-, 3-gram and skippy 2-gram with #topics = 15.

Figures 22–18 show 2D views on t-SNE 3D map (perplexity = 335) for cases for 1-, 3-gram and skippy 2-gram

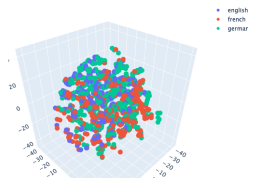

Figure 13 t-SNE 3D [#topics: 5; 1-gram; perplexity: 5]

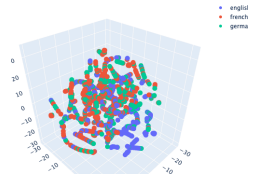

Figure 14 t-SNE 3D [#topics: 5; 3-gram; perplexity: 5]

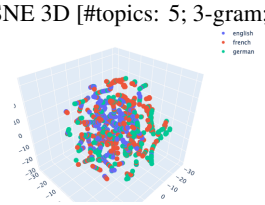

Figure 15 t-SNE 3D [#topics: 5; skippy 2-gram; perplexity: 5]

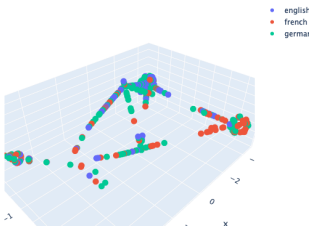

Figure 16 t-SNE 3D [#topics: 5; 1-gram; perplexity: 335]

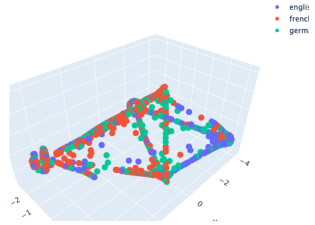

Figure 17 t-SNE 3D [#topics: 5; 3-gram; perplexity: 335]

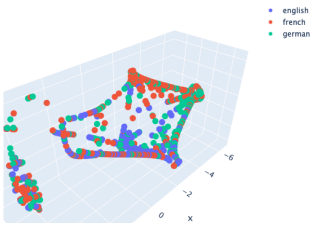

Figure 18 t-SNE 3D [#topics: 5; skippy 2-gram; perplexity: 335]

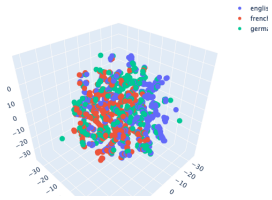

Figure 19 t-SNE 3D [#topics: 15; 1-gram; perplexity: 5]

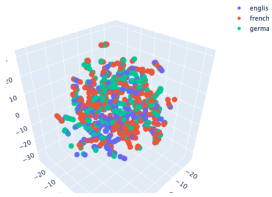

Figure 20 t-SNE 3D [#topics: 15; 3-gram; perplexity: 5]

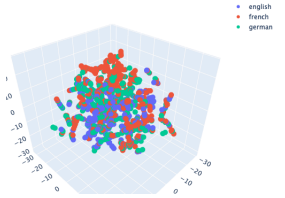

Figure 21 t-SNE 3D [#topics: 15; skippy 2-gram; perplexity: 5]

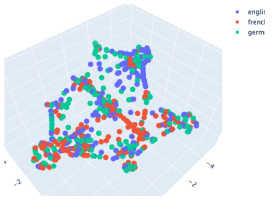

Figure 22 t-SNE 3D [#topics: 15; 1-gram; perplexity: 335]

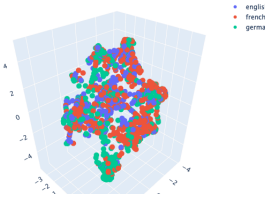

Figure 23 t-SNE 3D [#topics: 15; 3-gram; perplexity: 335]

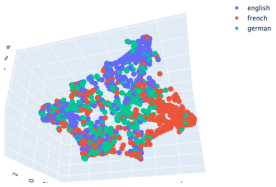

Figure 24 t-SNE 3D [#topics: 15; skippy 2-gram; perplexity: 335]

with #topics = 15.

4.3 Discussion

Let us examine cases with #topics = 5 first. With smaller perplexity values such as 5, clustering results do not differ qualitatively, yet they differ with larger values such as 335, as shown by the contrast among Figures 13, 14 and 15.

With larger perplexity values such as 335 clustering results differ qualitatively, and dimension reduction looks successful. This is indicted by Figures 16, 17 and 18. The differences among 1-gram, 3-gram and skippy 2-gram are basically smoothness of encoding: 3-gram and skip 2-gram based LDA capture richer structures than 1-gram base one.

It is hard to tell which of 3-gram or skippy 2-gram gives better results, but it is clear that skippy 2-gram based encoding can capture more complex structures.

Let us turn to cases with #topics = 15. Like cases with #topics = 5, clustering results do not differ qualitatively with smaller perplexity values such as 5. This is shown by the contrast among Figures 19, 20 and 21. Differences arise with larger perplexity values such as 335, but this time with Figures 22, 23 and 24, unlike cases with #topics = 5, it is harder to tell how they differ. This is probably because more number of topics give more degrees of similarity, or rather precisely more connectivities, to data points under analysis. This is why connection among data points look smoother in skip 2-gram and 3-gram settings.

Under this remark, though, it is not unreasonable to observe that Figure 24 gradually and most effectively localizes the distribution of Germanic and Romance sound patterns.

5 Conclusions

LDA is applied to encode spellings and pronunciations of a few languages and the encodings thus obtained were compared. English, French, German, Russian and Swahili were compared for spelling; English, French and German for pronunciation. Though the results at hand are rather preliminary, they still suggest that this unsupervised method successfully capture both similarities and dissimilarities among a set of languages examined, thereby revealing hidden regularities shared among them.

The virtue of this method is that we can make target data as large as we want, because data preparation is virtually cost-free. This research has targeted English, French, German, Russian, and Swahili, only because the submission deadline did not allow to add more languages.

In this paper, I deliberately forbid myself to explicate important insights gained from the obtained results, judging this is not the right place to do so.

Acknowledgements

To run `LDA tuning`, `R` (https://www.R-project.org/) version 4.3.x, developed by R Core Team, was used. To run `t-SNE`, the `manifold` module in `scikit-learn` (https://scikit-learn.org/) was used. For other data analysis and visualizations, Anaconda 3 (https://www.anaconda.com) version 23.11.00 was used, running Jupyter Notebook 6.5.4 on Python 3.10.

References

[1] 黒田航. 英単語の綴りと発音のズレの定量的評価. 認知科学会第 40 回大会発表論文集, pp. 644–647, 2023.

[2] D. Gutherie, B. Allison, W. Liu, L. Gutherie, and Y. Wilks. A closer look at skip-gram modeling. In **Proceedings of the 5th International Conference on Language Resources and Evaluation**, pp. 1–4, 2006.

[3] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. **Journal of Machine Learning Research**, Vol. 9, pp. 2579–2605, 2008.

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. **Journal of Machine Learning Research**, Vol. 3, pp. 993–1022, 2003.

[5] 岩田具治. トピックモデル. 講談社, 2015.

6 Appendix

Hierarchical clusterings of subsampled spells and sounds

Hierarchical clustering of 140 spells (subsamples) is shown in Figure 25. Leaf colors correspond to the colors assigned to the languages in Figures 1–12.

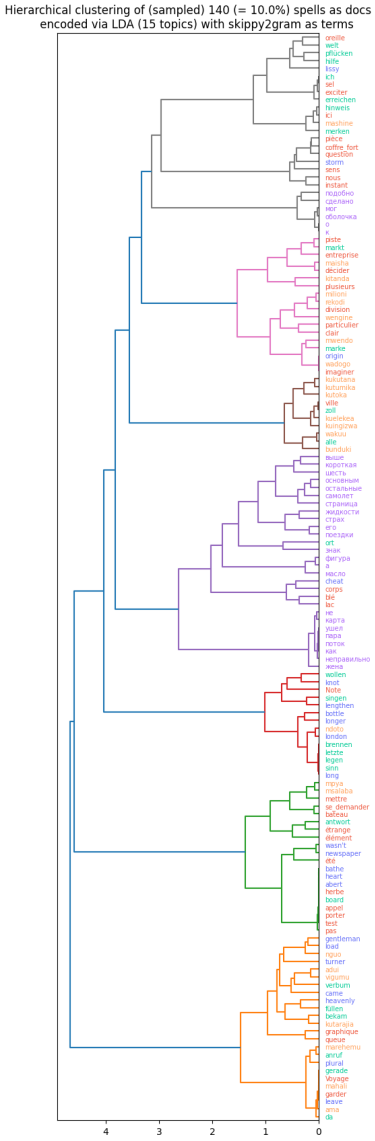

Figure 25 Hierarchical clustering of 140 sample spells encoded via skippy 2gram-based LDA (#topic: 15)

Hierarchical clustering of 166 sounds (subsamples) is shown in Figure 26. Leaf colors correspond to the colors assigned to the languages in Figures 13–24.

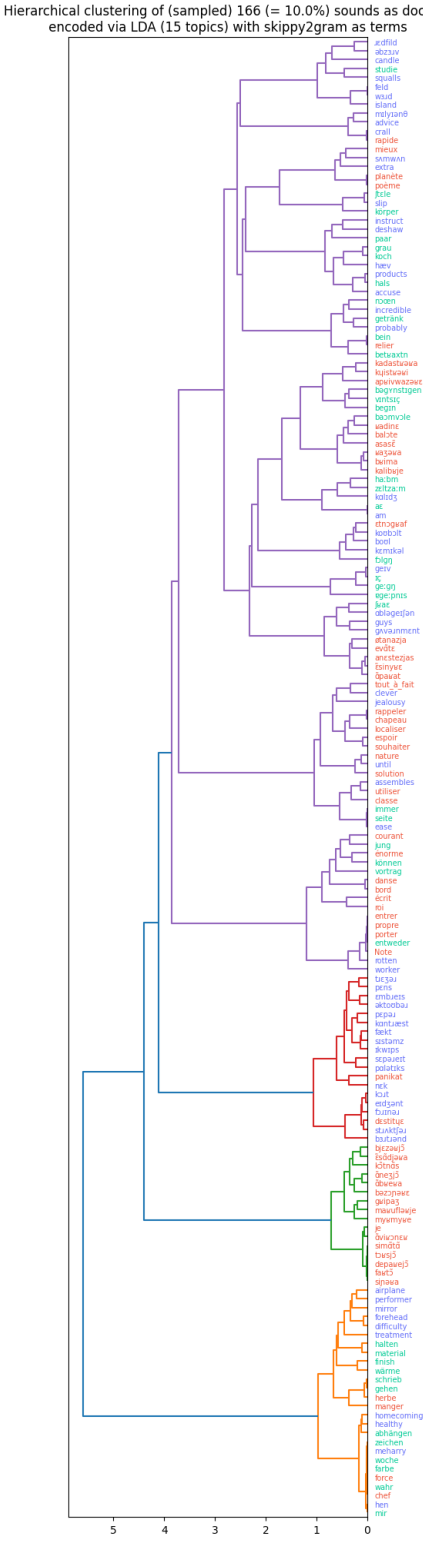

Figure 26 Hierarchical clustering of 166 sample sounds encoded via skippy 2gram-based LDA (#topic: 15)