Zero-shot query generation for approximate search algorithm evaluation

Pine, Aidan; Huggins-Daines, David; Leeming, Carmen; Littell, Patrick; Montler, Timothy; Souter, Heather; Turin, Mark

Publisher's version / Version de l'éditeur:

Proceedings of the Eight Workshop on the Use of Computational Methods in the Study of Endangered Languages, pp. 65-73, 2025-03-04

NRC Publications Archive Record / Notice des Archives des publications du CNRC :
https://nrc-publications.canada.ca/eng/view/object/?id=9fac9e31-71c3-48c3-98d6-9a46f121f374
https://publications-cnrc.canada.ca/fra/voir/objet/?id=9fac9e31-71c3-48c3-98d6-9a46f121f374

Zero-Shot Query Generation for Approximate Search Algorithm Evaluation

Aidan Pine[1], David Huggins-Daines[2], Carmen Leeming[2], Patrick Littell[1], Timothy Montler[3], Heather Souter[4], Mark Turin[5]

[1]National Research Council Canada, [2]Independent Researcher, [3]University of North Texas, [4]University of Winnipeg, [5]University of British Columbia

Correspondence: aidan.pine@nrc-cnrc.gc.ca

Abstract

Approximate search is a valuable component of online dictionaries for learners, allowing them to find words even when they have not fully mastered the orthography or cannot reliably perceive phonemic differences in the language. However, evaluating the performance of different approximate search algorithms remains difficult in the absence of real user queries. We detail several methods for generating synthetic queries representing various user personas. We then compare the performance of several search algorithms on both real and synthetic queries in two Indigenous languages, SENĆOŦEN and Michif, that are phonologically and morphologically very different from English.

1 Introduction

Online dictionaries are one of the most commonly used and important tools in language revitalization and reclamation programs (Anderson, 2020; Leavitt, 2023; Lyon et al., 2023). For under-resourced languages, online dictionaries are very often the only lexical resource available to learners in a community where no print dictionary has ever been compiled or published. For authoritative monolingual dictionaries, such as the Oxford English Dictionary, users are assumed to be fluent and literate in the language of the dictionary. The same expectations of users of bilingual dictionaries and phrasebooks in language revitalization contexts cannot be made. Users of bilingual dictionaries are often learners, and trying to harness the power of an online dictionary can present learners with an unwelcome paradox: they may wish to look up a word in the dictionary in order to learn it and/or verify the spelling, but in order to look it up in a dictionary with only an exact-match search algorithm, they already need to know exactly how to spell it. This can lead to a Catch-22, particularly with complex writing systems for which keyboard input systems are less standardized or easily available. For these reasons, it is extremely important in a language learning context that users can benefit from fuzzy search algorithms that accommodate anticipated errors or idiosyncratic spellings.

Despite the importance of online dictionaries in language revitalization, they remain resource-intensive endeavours that are often the first project that communities and scholars start and the last one to be completed (Sear and Turin, 2021; Schreyer and Turin, 2023). Compiling lexicographic data, let alone managing and maintaining software, websites and mobile apps all present significant technical hurdles (Trotter et al., 2023). On top of these requirements, building a language-specific approximate search algorithm is also a significant challenge. In some cases, language models already exist for the language in question and can be applied to provide morphologically-aware search results (Johnson et al., 2013; Arppe et al., 2021). Alternatively, Littell et al. (2017) describe software that allows users to define language specific phonologically-aware approximate search algorithms. Originally published under the name Waldayu, the software was generalized and renamed 'Mother Tongues Dictionaries' (MTD) in 2018. MTD is a Python library and collection of visualization frameworks that, given the MTD data specification, allows users to create online dictionaries (web, Android, iOS) from a potentially heterogeneous set of data (i.e., a spreadsheet, JSON file, and XML file). In addition to the data wrangling and visualization capabilities of the library, it also allows users to customize an approximate search algorithm based on weighted or unweighted edit distances. MTD has been used to develop online dictionaries for dozens of Indigenous languages around the world including in Canada, the US, and Japan.

Since 2017, the MTD search algorithm has been updated to allow multi-word, multi-field search,

Proceedings of the Eight Workshop on the Use of Computational Methods in the Study of Endangered Languages, pages 65–73

and to also include a multi-field variant of the BM25 ranking algorithm (Zaragoza et al., 2004) as a secondary score in addition to edit distance. Impressionistically, and through gathering informal user feedback, we believe that the changes to the MTD search algorithm have led to improved search results, although this has not been formally investigated. Part of the difficulty in evaluating approximate search is that a corpus of common misspellings or otherwise plausible queries does not exist for the Indigenous languages that we are working with. In this paper, we demonstrate a variety of techniques for generating plausible queries that can be applied to other written and unwritten languages. We then apply these techniques to dictionary data in the SENĆOŦEN and Michif languages and show that the recent updates to the MTD search algorithm provide improvements for each type of query generation strategy that we test. We believe the query generation techniques that we describe could be applied to other languages and used in other contexts to help evaluate approximate search algorithms.

2 Methodology

2.1 Data

To investigate a variety of approximate search algorithms, we apply our proposed query generation techniques to two lexical resources from two different languages; SENĆOŦEN and Michif.

2.1.1 SENĆOŦEN Dictionary

The SENĆOŦEN language is a Salish language spoken traditionally in the territories of the W̱SÁNEĆ people. Contemporary revitalization efforts were catalyzed by the late Dave Elliott Sr., who developed the SENĆOŦEN orthography which is still the standard used by the community. The SENĆOŦEN dictionary (Montler, 2018) is the largest lexicographic resource available for the SENĆOŦEN language, and contains over 30 000 words and example sentences. From an approximate search perspective, the language is particularly challenging given its rich and complex phoneme inventory. The language has 36 different consonants with phonemic contrasts between velar and uvular consonants as well as rounding and glottalization. These contrasts are all represented in the orthography, often only with small diacritical changes to indicate them, as illustrated in Table 1.

K̲ /q/	Ḱ /q^{w}/	K /q'/
K̓ /q^{w}'/	Q /k^{w}'/	Ȼ /k^{w}/

Table 1: A subset of the SENĆOŦEN consonant inventory illustrating how uvular/velar, rounding, and glottalization contrasts are encoded in the orthography. This phonological richness presents a challenge for learners when searching in the dictionary.

2.1.2 Michif Dictionary

Southern Michif is one of three language varieties spoken by the Métis (Bakker, 1997; Sammons, 2019). It is a contact language combining elements from Algonquian languages—Plains Cree and the Saulteaux dialect of Ojibwe—with Métis French. Traditionally, it has been written with a mixture of English and French spelling conventions, notably as seen in the Turtle Mountain Dictionary (Laverdure et al., 1983) and its recent digital version (Souter et al., 2024b). More recently there has been an effort to further develop and use the Southern Michif Learners Orthography which is based on a double-vowel system similar to that used for Ojibwe. It has its roots in the work done initially by the late Rita Flamand of Camperville, Manitoba with later input from Robert Papen. Further refinement was carried out by a number of learners. And, after reflection on the early work of Ida Rose Allard, a decision was made to use one special symbol, ñ, to mark nasalization of preceding vowels in order to help support accurate pronunciation.

The Southern Michif for Learners website (Souter et al., 2024a) includes an extensive set of illustrated phrases and words with audio recordings, used with permission here.

2.2 Evaluation

To evaluate the performance of our search strategies we employ the use of mean reciprocal rank (MRR). MRR is a measure for evaluating the order of results given a query. Concretely for our use case, if we type a query and the dictionary entry we intended to find is ranked first in the search results then it has a reciprocal rank of 1 (which denotes the best possible score); however if the expected entry does not rank at all in the search results then the reciprocal rank is 0 (which denotes the worst possible score). If the expected dictionary entry appears as the second result, it would have a recip-

rocal rank of $\frac{1}{2}$ (and $\frac{1}{3}$ if it appeared as the third result, etc). The mean reciprocal rank is then the mean of each reciprocal rank for each query that we evaluate.

More formally, we calculate MRR as $\frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{r_i}$ where Q is a set of queries and r_i is the rank of the expected entry from the dictionary for the *i*-th query. So, in order to calculate this metric, we need a set of queries Q and a corresponding set of expected dictionary entries E.

One can imagine obtaining Q from actual queries logged from an online dictionary - the expected dictionary entries E could also then be obtained by asking users to select the entry they were looking for if it appeared in their search results. This approach is somewhat noisy though, and more importantly, is incompatible with the privacy terms of the dictionaries we use. As a rule, we do not log open user input since even when this information is anonymized, there is no guarantee that users will not input de-anonymized or sensitive search terms (e.g., searching for one's own name or other identifying features). Furthermore, these dictionaries often operate entirely offline which would complicate our ability to record these results.

Instead, this paper presents eight methods for *approximating* user queries given a set of dictionary entries. Since some of our methods are time-consuming, we limit our evaluation to a randomly-sampled 50-word subset from each dictionary. That is, we apply each of the eight methods discussed in the following section to a randomly sampled subset of the dictionaries. We then use mean reciprocal rank to evaluate how robust the search algorithms discussed in §3 are with respect to the approximated user queries.

Our approach here has the simplifying assumption that there is only one expected entry for any given generated query. In reality, there are often multiple relevant entries given a query, for example morphologically related words or matches found in other fields related to the main entry (e.g., an example sentence). If we were able to accurately identify all relevant entries in the dictionary for a particular query, we might instead have considered evaluating using mean average precision.

2.3 Generating queries

To help guide the creation of our query generation functions, we borrow descriptions of likely users from Littell et al. (2017). The sections 2.3.1 to 2.3.4 describe a variety of different types of users and our corresponding query generation techniques. To further approximate the types of queries made by a learner, we consider additional approaches to query generation in sections 2.3.5 to 2.3.8.

2.3.1 Users who can distinguish phonemes but do not always know the orthographic conventions

In order to generate queries for this category of user, the first and second authors hand-transcribed words in the target language by listening to audio of those words. Both transcribers were familiar with the sound systems of the languages they were transcribing, but were not speakers, and had not had any instruction in the language or its writing system. They simply listened to the audio with headphones and transcribed what they heard using the (non-IPA) keyboard available to them. Both transcribers are first-language English speakers who also speak French and have formal training in linguistics.

2.3.2 Users who know the orthography, but cannot reliably discern certain phonemes

We approach query generation for this category of user as a data corruption task. We target specific classes of graphemes and phonemes that we expect to be challenging for our users to distinguish (i.e., the velar/uvular contrast in SENĆOŦEN). We then randomly corrupt up to $N = 3$ of these phonemes' related graphemes with another confusable from the same class, for example swapping out Ḱ (/qʷ/) for Ḳ (/qʷ'/) in SENĆOŦEN or ñ (indicating nasalization on the preceding vowel) for n (/n/) in Michif.

2.3.3 Users without access to a keyboard

We assume this category of user to be able to accurately identify and discern phonemes in the target language, and to also be familiar with the target language's writing system, but to be unable to type due to the unavailability of a Unicode input system on their device. This is less of an issue for Michif, but for SENĆOŦEN there are many specialized Unicode characters and diacritics used in the writing system, and typing in the language requires installing a language-specific keyboard (Chase and Borland, 2022).

To approximate the type of user queries expected when such a keyboard is not installed, we transform each non-ASCII character to its closest ASCII

equivalent. We do this by performing NFD Unicode normalization, removing any diacritics in the range U+0300 to U+036F, and then applying the Unidecode[1] library to the resulting text.

2.3.4 Users who know an alternative orthography

For this class of user queries, we generate queries for Michif in the Turtle Mountain Dictionary (TMD) orthography, an alternate orthography to the one used in the Michif dictionary in this study. SENĆOTEN has historically had multiple orthographies, including an Americanist phonetic representation and the Bouchard practical orthography (see Turner and Hebda (2012, p. 155)). These orthographies are not in standard use by the community and while they might still be used in some queries of the SENĆOTEN dictionary, it would be uncommon, and we do not include them.

2.3.5 IPA-based query generation

The vast majority of users of the SENĆOTEN dictionary are first-language English speakers. This is similar for Michif except in some cases users might speak French as a first language. This query generation technique seeks to approximate a query by mapping it through the International Phonetic Alphabet (IPA) to a query language such as English.

First, we use a rule-based grapheme-to-phoneme (G2P) library (Pine et al., 2022) to derive the IPA pronunciation form of a given word in the dictionary. We then use PanPhon (Mortensen et al., 2016) to map from the IPA symbols in the target language, to the closest English IPA equivalents. For Michif, we also map to the closest French IPA equivalents for comparison, since speakers of that language are more likely to speak French as a first language.

Finally, we train two sequence-to-sequence Transformer based models using the DeepPhonemizer[2] software. We train an English system using a reversed IPA version of the CMU pronunciation dictionary[3] to predict IPA from English graphemes, and we train a French system in the same way using the WikiPronunciation dictionary[4]. In both cases we keep the default hyperparameter settings and train until convergence (140k steps for English with a 12% character error rate (CER), and 1700k steps for French with a 10% CER).

We release our English[5] and French[6] phoneme-to-grapheme models publicly. For generating plausible English or French queries from another language, a method of turning graphemes into phonemes in the target language would be required. For generating plausible queries in a language other than English or French, a similar phoneme-to-grapheme model in that language would also have to be trained.

2.3.6 LLM-based query generation

We also consider the use of Large Language Models (LLM) as naive transcribers of the languages in question. We use ollama and the publicly available 'llama3' model. For the Michif prompts, we ask the question 'The following is a list transcriptions of words in the Michif language. How would you write these words using only the English or French orthography?' and for SENĆOTEN we prompt it to only write the words using the English orthography. We then provide the list of IPA transcriptions of Michif or SENĆOTEN words and record the results. Like the previously described method, adapting this approach for other languages would also require providing the LLM with a pronunciation form of the words in question.

2.3.7 Audio-based query approximation (ASR)

Instead of approximating user queries through a transformation of the original text, we also consider approximating user queries by decoding the original audio using automatic speech recognition (ASR) models. To mimic how a user of the dictionary 'with English ears' might transcribe a word, we decode audio corresponding to a given query with the pre-trained wav2vec2-base-960h model[7]. Importantly, we use a greedy decoder that is not constrained by a language model, so the model will decode the audio into characters in the English orthography, but will allow for non-English words to be decoded. While the Michif dictionary had audio available for each of the 50 words that were sampled, the SENĆOTEN did not, so we synthesized the audio using the SENĆOTEN speech synthesis model described in Pine et al. (2025).

[1] https://pypi.org/project/Unidecode/
[2] https://github.com/as-ideas/DeepPhonemizer
[3] https://github.com/open-dict-data/ipa-dict
[4] https://github.com/DanielSWolf/wiki-pronunciation-dict
[5] https://bit.ly/eng-p2g-model
[6] https://bit.ly/fra-p2g-model
[7] https://huggingface.co/facebook/wav2vec2-base-960h

2.3.8 Teacher-curated queries

It is difficult to draw any conclusions from artificially generated queries. Part of the problem is that for each word in the dictionary, there are many ways to misspell it. So, we cannot know if the ways our query generation techniques have misspelled these terms are similar to the way the target audience of these dictionaries will misspell them.

To help corroborate the results seen among our query generation techniques, the third, fifth and sixth authors, who have experience in teaching Michif and SENĆOŦEN and are familiar with common misspellings from students, compiled a list of common misspellings for each of the words in the 50-word subsets of the Michif and SENĆOŦEN dictionaries.

2.4 Examples and CERs of each technique

In Table 2 we show the result of applying each method to one of the words in each dictionary.

Query Type	Michif	SENĆOŦEN
Original	pashikook	NEW̱SPEŦTENEK̲
IPA	pʌʃɪkoːk	nəxʷspəs̟tə́nəq
Human (§2.3.1)	peshkop	nuhuhspahstanak
Phon. (§2.3.2)	pawshiihkok	NEW̱SPEŦTENEK̶
ASCII (§2.3.3)	pashikook	NEWSPETTENEK
P2Eng (§2.3.5)	puchikouk	neckspothtinick
P2Fra (§2.3.5)	péchecauque	nekspestenek
LLM (§2.3.6)	Pashikotak	Nexwspetheniq
ASR (§2.3.7)	PUSHCOG	NOSPASTANA
Teacher (§2.3.8)	pashikohk	NEW̱SPESTENEK̲

Table 2: An example of how a sample word in each dictionary is transformed by each query generation method. Each example here is the raw output from each query generation method (i.e., prior to case normalization).

As mentioned in §2.2, to generate our test set, we randomly sample a 50-word subset from each dictionary. We then apply each proposed query generation method to the 50-word test set. In Table 3 we report the character error rate (CER) between the generated queries and the original terms. Note that this is not an evaluation of the query generation technique (which we cannot do without data of actual misspelled words and a model of their distribution), rather it is just meant to be an indication of how much the generated queries deviate from the original terms. A higher CER indicates an increased difficulty for the task of approximate search, but not necessarily a less valid or less plausible query.

Across the board, our query generation techniques incurred higher CERs in SENĆOŦEN than they did for Michif. For example, the ASCII query generation technique (§2.3.3) incurs a 30% CER for SENĆOŦEN but only a 1% CER for Michif. In other words, for SENĆOŦEN, non-ASCII characters make up 30% of the characters in our 50-word set, whereas they only make up 1% of the characters in our set for Michif.

Query Type	Michif	SENĆOŦEN
Human (§2.3.1)	0.37	1.38
Phon. (§2.3.2)	0.52	0.27
ASCII (§2.3.3)	0.01	0.30
P2Eng (§2.3.5)	0.43	1.18
LLM (§2.3.6)	0.34	1.01
ASR (§2.3.7)	0.59	0.81
Teacher (§2.3.8)	0.40	0.29

Table 3: Query generation Methods and their Character Error Rates (CER). CERs in terms of the character edit distance between the words generated by the query generation method, and the terms in the dictionary they are meant to approximate.

2.5 Adapting to other languages

Beyond evaluating the recent changes to the MTD search algorithm, part of the goal of this paper is to provide query generation techniques that can be applied to languages other than SENĆOŦEN and Michif. Table 4 shows the data or models required to implement each technique, since some techniques require only audio and some techniques require text, or an available grapheme-to-phoneme library for the language in question.

Query Type	G2P	Audio	Text
Human (§2.3.1)	✗	✓	✗
Phon. (§2.3.2)	✗	✗	✓
ASCII (§2.3.3)	✗	✗	✓
P2Eng (§2.3.5)	✓	✗	✓
LLM (§2.3.6)	✓	✗	✓
ASR (§2.3.7)	✗	✓	✗

Table 4: Query generation Methods and their requirements. ‘G2P’ indicates that the method requires a grapheme-to-phoneme engine to be adapted to a new language.

3 Search Algorithms

Following Littell et al. (2017) we compare results using both an unweighted Levenshtein edit distance $ULev$ and a weighted Levenshtein edit distance $WLev$. The unweighted Levenshtein edit distance

between two strings X and Y is equal to the number of single-character edits (additions, deletions, substitutions) required to change X into Y. By comparison, the weighted Levenshtein edit distance allows edits to be weighted differently, for example allowing substitutions involving commonly confused characters to accrue a lesser penalty. We used the hand-written substitution weights that have been in use for the dictionaries already.

In addition to ranking results based on edit distance, the most recent version of the MTD search engine also applies a secondary ranking based on a weighted multi-field variant of BM25 (Zaragoza et al., 2004); a language agnostic ranking function based on the inverse document frequency of the query. Therefore, in addition to evaluating the difference between weighted and unweighted edit distance, we also report the effect of including BM25 as a secondary score. Although MTD is capable of handling multi-word queries and indexing multiple fields, for the purposes of this evaluation we limit ourselves to single word queries and only search based on a single field in the dictionary entries. The MTD search engine also allows for optional stemming when creating the inverted index used in searching, as well as some basic normalization functions including case and Unicode normalization and the removal of punctuation. These configurations result in the same normalization processes being applied to each term in the inverted index and to each query. For the purposes of this paper we do not configure a stemmer, but we do apply both case and Unicode normalization to all of the queries and to each term in the inverted indices built by MTD.

4 Results

To evaluate the approximate search algorithms described in §3, we randomly sample 50 words from each of the dictionaries. We then apply each of our query generation techniques to the random 50-word sample sets for both languages and compute the mean reciprocal rank (MRR) for the queries generated by each technique. We present our results in Table 6 on the following page.

As expected, given the wide range of CERs for our various query generation techniques, there is also a wide range of results and the relationship between CER and MRR appears roughly inverse. For example the P2Eng (§2.3.5) technique, which had a CER of 1.18, only receives a MRR of 0.07 in the best system for SENĆOŦEN while the ASCII system for Michif had a 0.01 CER and resulted in a MRR of 0.96 in the best systems.

The addition of BM25 results in MRR improvements across all query generation strategies for both weighted and unweighted edit distance. We also see improvements to the MRR when BM25 is included for unmodified queries. That is, when we pass the original word unchanged as the query, we see improvements of +0.09 MRR for SENĆOŦEN and +0.15 MRR for Michif as well as improvements among all query generation techniques. We believe that this is sufficient for justifying the use of an approximate search strategy that is combined with BM25, like the one found in MTD.

To weight or not to weight The difference between weighted and unweighted edit distance is less clear than the improvements seen with the addition of BM25. In Table 5 we compare the results when prompting the LLM to produce either English or French outputs, as well as mapping through English or French pronunciation forms for the phoneme-to-grapheme based technique (§2.3.5). Unexpectedly, the results from the English LLM and P2Eng methods do not seem to show a strong difference between weighted and unweighted edit distances whereas we see a stronger improvement for the generated 'French' queries using an unweighted edit distance. Since the Michif dictionary substitution weights were written by a first-language English speaker who works primarily with English-speaking students, the pattern that we see here could be the result of linguistic bias in the substitution weights, which could be either desirable or undesirable depending on the target audience for the dictionary. In this case, it is possible that the weights are resulting in worse performance for French-influenced queries, since the weights were created with an English speaking audience

	MRR ↑	
Query Type	MTD_w	MTD_u
P2Eng	0.39	0.41
P2Fra	0.18	**0.38**
LLM Eng	0.66	0.65
LLM Fra	0.30	**0.39**

Table 5: Mean Reciprocal Ranks (MRR) for Michif IPA-based (§2.3.5) and LLM (§2.3.6) query generation with both English and French outputs. CER denotes the Character Error Rate for the 50 word set for each particular query generation technique.

Query Type	Language	CER	MRR ↑ $ULev$	$WLev$	MTD_w	MTD_u
Original Text	SENĆOŦEN	0.0	0.91	0.91	1.0	1.0
Phon. (§2.3.2)	SENĆOŦEN	0.27	0.63	0.54	0.58	**0.68**
Teacher (§2.3.8)	SENĆOŦEN	0.29	0.09	0.09	0.11	0.12
ASCII (§2.3.3)	SENĆOŦEN	0.30	0.35	0.36	**0.45**	0.42
ASR (§2.3.7)	SENĆOŦEN	0.81	0.01	0.03	0.04	0.03
LLM (§2.3.6)	SENĆOŦEN	1.01	0.06	0.10	**0.14**	0.11
P2Eng (§2.3.5)	SENĆOŦEN	1.18	0.02	0.04	0.06	0.07
Human (§2.3.1)	SENĆOŦEN	1.38	0.0	0.01	0.02	0.0
Original Text	Michif	0.0	0.80	0.81	0.96	0.96
Phon. (§2.3.2)	Michif	0.52	0.33	0.33	0.41	**0.44**
Teacher (§2.3.8)	Michif	0.40	0.47	0.48	0.61	0.60
ASCII (§2.3.3)	Michif	0.01	0.79	0.79	0.96	0.96
ASR (§2.3.7)	Michif	0.59	0.12	0.12	0.15	**0.20**
LLM (§2.3.6)	Michif	0.34	0.46	0.52	0.66	0.65
P2Eng (§2.3.5)	Michif	0.43	0.25	0.26	0.39	0.41
Human (§2.3.1)	Michif	0.37	0.43	0.42	0.50	**0.55**
TMD Queries (§2.3.4)	Michif	0.79	0.18	0.26	**0.30**	0.25

Table 6: Mean Reciprocal Ranks (MRR) for different query generation techniques given 50 randomly sampled words from the SENĆOŦEN and Michif dictionaries. CER denotes the Character Error Rate for the 50 word set for each particular query generation technique. MTD indicates the search strategy used by Mother Tongues Dictionaries ranks results based on edit distance and a secondary BM25 score. A higher MRR for a particular search strategy indicates that it is more robust to that type of query.

in mind. Ultimately, we believe that the decision of whether to use substitution weights should depend on how well the target audience is known in advance. In most cases though, given the time and expertise required to create custom substitution weights for each language, unweighted edit distance is likely sufficient.

5 Conclusion

In this paper, we have proposed and developed a variety of methods for approximating user queries. We provide guidance and release models so that they might be adapted to other languages. Using the described query generation techniques, we compared the effectiveness of a variety of approximate search algorithms in both SENĆOŦEN and Michif dictionaries. We showed that fuzzy search can be improved by combining BM25 as a secondary score with Levenshtein edit distance. Despite these improvements, and the relatively successful results for Michif, approximate search remains a difficult problem, particularly for languages with large phoneme inventories like SENĆOŦEN.

Future work should compare the queries generated using our described techniques with actual misspellings, for example using corpora like the ones described in Max and Wisniewski (2010) and Flor et al. (2019). Additional future work could also more thoroughly explore the difference between weighted and unweighted edit distances for example by applying the methods described here to more languages, or by devising techniques for learning optimal edit distance weights from data. The latter approach would require a corpus of real or artificial misspelled data, as well as careful evaluation to avoid over-fitting to the training data.

Additional future work might also consider morphologically-aware query generation and approximate search algorithms, for example comparing the FST-based morphologically-aware approaches of Johnson et al. (2013) with the phonologically motivated techniques described here. We expect that languages with higher degrees of polysynthesis might in turn require search algorithms with greater morphological awareness, but it is not clear at what point the benefits of morphologically aware search would be large enough to motivate the additional effort compared to, for example, a simple unweighted edit distance in combination with a secondary BM25 score.

Acknowledgments

This work would not have been possible without the support from our collaborators at the W̱SÁNEĆ School Board, PENÁĆ, SX̱EDŦELISIYE, and Tye Swallow. We would also like to thank Delaney Lothian, Maria Ryskina, and Roland Kuhn for proof-reading and assistance formatting and typesetting this document.

References

Patricia Anderson. 2020. *Revitalization Lexicography: The Making of the New Tunica Dictionary*. University of Arizona Press, Tuscon.

Antti Arppe, Jolene Poulin, Eddie Antonio Santos, Andrew Neitsch, Atticus Harrigan, Katherine Schmirler, Daniel Hieber, Ansh Dubey, and Arok Wolvengrey. 2021. Towards a morphologically intelligent and user-friendly on-line dictionary of Plains Cree—next next round.

Peter Bakker. 1997. *A Language of Our Own : The Genesis of Michif, the Mixed Cree-French Language of the Canadian Metis*. Oxford University Press, Oxford & New York.

Bridget Chase and Kyra Borland. 2022. Networks of support: How online resources are built, maintained, and adapted for community language revitalization needs at firstvoices. *Language Documentation & Conservation*, pages 209–227.

Michael Flor, Michael Fried, and Alla Rozovskaya. 2019. A benchmark corpus of English misspellings and a minimally-supervised model for spelling correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 76–86, Florence, Italy. Association for Computational Linguistics.

Ryan Johnson, Lene Antonsen, and Trond Trosterud. 2013. Using Finite State Transducers for Making Efficient Reading Comprehension Dictionaries. *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013)*, pages 59–71.

Patline Laverdure, Ida Rose Allard, and John C. Crawford. 1983. *The Michif Dictionary: Turtle Mountain Chippewa Cree*. Pemmican Publications, Winnipeg.

Robert Leavitt. 2023. Creating the Passamaquoddy-Wolastoqey dictionary: A personal reflection on fifty years of lexicography. *Dictionaries: Journal of the Dictionary Society of North America*, 44:187–206.

Patrick Littell, Aidan Pine, and Henry Davis. 2017. Waldayu and Waldayu Mobile: Modern digital dictionary interfaces for endangered languages. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 141–150, Honolulu. Association for Computational Linguistics.

John Lyon, Justine Manuel, and Kathleen Michel. 2023. The Upper Nicola Nsyilxcn talking dictionary project: Community-driven revitalization lexicography within an academic context. *Dictionaries: Journal of the Dictionary Society of North America*, 44:107–126.

Aurélien Max and Guillaume Wisniewski. 2010. Mining naturally-occurring corrections and paraphrases from Wikipedia's revision history. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Timothy Montler. 2018. *SENĆOŦEN: A Dictionary of the Saanich Language*. University of Washington Press, Seattle, WA, USA.

David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. PanPhon: A resource for mapping IPA segments to articulatory feature vectors. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484, Osaka, Japan. The COLING 2016 Organizing Committee.

Aidan Pine, Erica Cooper, David Guzmán, Eric Joanis, Anna Kazantseva, Ross Krekoski, Roland Kuhn, Samuel Larkin, Patrick Littell, Delaney Lothian, Akwiratékha' Martin, Korin Richmond, Marc Tessier, Cassia Valentini-Botinhao, Dan Wells, and Junichi Yamagishi. 2025. Speech generation for Indigenous language education. *Computer Speech & Language*, 90.

Aidan Pine, Patrick Littell, Eric Joanis, David Huggins-Daines, Christopher Cox, Fineen Davis, Eddie Antonio Santos, Shankhalika Srikanth, Delasie Torkornoo, and Sabrina Yu. 2022. G_i2P_i rule-based, index-preserving grapheme-to-phoneme transformations. In *Proceedings of the Fifth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 52–60, Dublin, Ireland. Association for Computational Linguistics.

Olivia Sammons. 2019. *Nominal Classification in Michif*. Ph.D. thesis, University of Alberta.

Christine Schreyer and Mark Turin. 2023. Indigenous lexicography: An introduction. *Dictionaries: Journal of the Dictionary Society of North America*, 44:1–5.

Victoria Sear and Mark Turin. 2021. Locating criticality in the lexicography of historically marginalized languages. *History of Humanities*, 6:237–259.

Heather Souter, Carmen Leeming, Marlee Paterson, and Terry Ireland. 2024a. *Southern Michif for Learners*. Prairies to Woodlands Indigenous Language Revitalization Circle.

Heather Souter, Olivia Sammons, and David Huggins Daines. 2024b. Creating digital learning and

reference resources for Southern Michif. In *Proceedings of the Seventh Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 67–75, St. Julians, Malta. Association for Computational Linguistics.

Bailey Trotter, Christine Schreyer, and Mark Turin. 2023. An open-access toolkit for collaborative, community-informed dictionaries. *Dictionaries: Journal of the Dictionary Society of North America*, 44:161–185.

Nancy Turner and Richard Hebda. 2012. *Saanich Ethnobotany: Culturally Important Plants of the W̱SÁNEĆ People*. Royal British Columbia Museum.

Hugo Zaragoza, Nick Craswell, Michael J. Taylor, Suchi Saria, and Stephen E. Robertson. 2004. Microsoft Cambridge at TREC 13: Web and hard tracks. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*, volume 500-261 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).