Proceedings of the 1st Joint SLTU and CCURL Workshop (SLTU-CCURL 2020), pages 28–35
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Two LRL & Distractor Corpora from Web Information Retrieval and a Small Case Study in Language Identification without Training Corpora

Armin Hoenen, Cemre Koc, Marc Rahn
Goethe University Frankfurt
Empirical Linguistiscs, Juridicum, Senckenberganlage 29, 60325 Frankfurt
hoenen@em.uni-frankfurt.de, cem_koc@icloud.com, marc.rahn@venturerebels.de

Abstract

In recent years, low resource languages (LRLs) have seen a surge in interest after certain tasks have been solved for larger ones and as they present various challenges (data sparsity, sparsity of experts and expertise, unusual structural properties etc.). For a larger number of them in the wake of this interest resources and technologies have been created. However, there are very small languages for which this has not yet led to a significant change. We focus here one such language (Nogai) and one larger small language (Māori). Since especially smaller languages often face the situation of having very similar siblings or a larger small sister language which is more accessible, the rate of noise in data gathered on them so far is often high. Therefore, we present small corpora for our 2 case study languages which we obtained through web information retrieval and likewise for their noise inducing distractor languages and conduct a small language identification experiment where we identify documents in a boolean way as either belonging or not to the target language. We release our test corpora for two such scenarios in the format of the An Crúbadán project (Scannell, 2007) and a tool for unsupervised language identification using writing system and toponym information.

Keywords: similar languages, less resourced languages, language identification, distractor languages, Māori, Nogai

1. Introduction

For Less Resourced Languages (LRLs), it may be especially hard to obtain data. The smaller the LRL is, the harder this will tendentially be (apart from some very well described small languages). The level and degree of possible expertise and the number of linguistic descriptions decreases. Thus, labelling and obtaining labelled data for these cases is especially hard and often unrealistic. Language Identification (LI) on the other hand uses mainly supervised methods with training corpora for which the language/s or variety/ies is/are known. This extends to the Discrimination between Similar Languages (DSL) task. In Web Information Retrieval (WIR) for LRLs LI can be part of a pipeline, be it in manual or automatic extraction. A retrieved document must be classified as relevant or not for an LRL corpus. Since labelled data is often not a priori available for the training of LI classifiers, in this paper, we present a very simple approach which leans only on resources which are relatively easily obtainable.

The paper is organized as follows: Section 2. recounts briefly the large body of related work. Section 3. then describes in detail how to define similar language scenarios (in WIR and beyond) by using linguistic criteria, before a classifier and its features are presented along the test scenario corpora. Section 4. summarizes the results of the main experiment, which are discussed alongside some detail on toponyms in Section 5. Finally, Section 6. briefly summarizes the achievements and concludes.

2. Related Work

In order to compose a noise-free corpus even for very small languages for which expertise (and thus the capacity of noise recognition in the face of similar sister languages) is very limited, we need a method to discriminate between the target language and what we want to call *distractors*. We understand this step as crucial for corpus generation.

This paper draws from two subfields, the first one being DSL as closely related to LI and the second one WIR. The task of LI precedes that of DSL, which has come up after standard language identification had been shown to work less well for similar languages (see for instance Padró and Padró (2004), Martins and Silva (2005),Ljubesic et al. (2007)). Tiedemann and Ljubešić (2012) developed methods for the efficient discrimination between Croatian, Serbian, Slovenian and Bosnian, Ranaivo-Malançon (2006) for Malay and Indonesian. In 2014, the first DSL shared task has been conducted (Zampieri et al., 2014) which has since been run so far until 2018 (Zampieri et al., 2018). Approaches to language identification and similar language discrimination have been plenty and Jauhiainen et al. (2019) give a recent overview. The large majority of these has used supervised techniques trained and tested on labeled data. As for unsupervised scenarios, clustering and other approaches have been used.

Our method is based on intersections of grapheme inventories. Henrich (1989) already use knowledge on peculiar letters in alphabets. Some other researchers also employed them in language identification in various ways (Giguet, 1995; Hanif et al., 2007; Samih and Kallmeyer, 2017; Hasimu and Silamu, 2017). Our binary intersection approach is to our best knowledge new as is the combination with place names. As for place names, for instance Chen and Maison (2003) have shown that place names can be successfully used in person name LI since their ngrams are more typical than those extracted from normal text.

WIR is a constantly active field since the seminal paper of *Web as Corpus* (Kilgarriff and Grefenstette, 2001). For LRLs, several works have been published (Biemann et al., 2007; Scannell, 2007) partly releasing publicly available repositories such as the Leipzig Corpora Collection. The retrieval of LRL content on the web is complicated by the fact that large parts of the web consist of content in the

largest languages[1] and that those matter most for the business models of large search engines. Scannell (2007) consequently speaks of 'polluting languages' when characterising unwanted results in LRL queries. We draw from such aspects of these studies as well as from general linguistic literature on language genealogy and contact phenomena (see for instance (Cysouw, 2013; Thomason, 2001)).

3. Method

In order to facilitate WIR in particular for LRLs, we present an approach to rigorously define similar language scenarios and implement a binary classifier for each pair *target language-distractor language* using writing system related and toponymic information.

3.1. Defining Distractors

Ljubesic et al. (2007) in her first paper discriminated Croatian, Serbian and Slovenian, then in a follow-up, the variety of Bosnian was included (Tiedemann and Ljubešić, 2012). This variety had only recently become ever more recognized as a language in the aftermath of the civil war in former Jugoslavia. However, as this example shows, apart from the fuzzy border between what can count as language and what as dialect (Barfield, 1998, p.85), other factors such as availability of labelled data or the official status of a variety may play a role when deciding which languages to include in a particular DSL task. Here, we advocate a linguistically informed uniform approach towards the definition of what we'd like to call *distractor languages*. We propose to take into account the following three types of target languages:

- language isolates without known relatives,
- pidgin and creole laguages, and
- all other languages.

Departing from this distinction, now distractors are languages which share confusingly many features with a target language. In the case of language isolates, fortunately there are no linguistically closely related sister languages, so only a language which has intense contact can be potentially confusing. This language (languages) can usually be identified by an analysis of language geography and history as well as loanwords. In WIR, we would argue that one should include this language in the distractors since it will often be a very ubiquitous web language, but for evaluating LI tasks alone, depending on the amount of loaning and the degree of orthographic adaptation this might not always be necessary.

For creoles obviously the superstrate language[2] is the most obvious distractor coming to mind. In case of English and French based creoles (which should be the most numerous)[3] however much orthographic simplification may apply rendering them somewhat less confusable with the superstrate. This however depends on the nature of the superstrate's writing system, which for English is especially deep (Katz and Frost, 1992). This can not be presupposed for the general case. Some linguists have hypothesized a universality of certain features of creoles such as double negation (Déprez and Henri, 2018) which would render them similar amongst each other apart from the often parallel simplification of the superstrate. Their common vocabulary will render them similar, too. So far, we have not found a DSL on the discrimination between creoles of the same superstrate although this could be a challenging task. However, for creoles, also the substrate language contributes fewer or more lexemes and should thus be considered on a case by case basis as possible distractor.

For all other languages, the most probably similar languages are closely related sister languages (written in the same script), where the degree of similarity correlates with the degree of relatedness (Cysouw, 2013). For those languages, one should thus adhere to a language genealogy, such as the ones provided by the WALS (Dryer and Haspelmath, 2013) or Ethnologue[4] and define those languages as distractors which are most closely related. In another experiment (Hoenen et al., 2020), we found that computing the overlap of most frequent words can provide useful hints to which languages from the same family (or sprachbund) should be included. We found in a scenario for Galician that Spanish and Portuguese clearly showed most lexical overlap followed by Italian with approximately a quarter of the similarity, then followed with a large gap indistinguishably by other Romance, Germanic and other languages. Historical stages of the target language – even in case it is an isolate – should always count as distractors.

For WIR, there might also be paralinguistic distractors such as badly OCRed text, written glossolalia, program code, cryptographic cyphers or other artifacts. Summarizing the approach to defining similar language discrimination scenarios:

- language isolates: contact languages (orthography, loanword sources)
- pidgin and creole laguages: other creoles based on the same superstrate language, superstrate language, substrate language
- all other languages: closely related languages, contact languages

Thresholds for similarity must be chosen according to the scenario on a case by case basis, since context (here WIR, writing systems), number of relatives and interrelatedness (if there are only 2 close relatives one may want to include

[1] As one can see from statistics on pages such as `https://w3techs.com/technologies/overview/content_language` or `https://www.internetworldstats.com/stats7.htm`, both last accessed on 30-03-2020.

[2] This is for instance the language of the former colonial power from which the creole has then inherited its vocabulary.

[3] A hint towards this is found when searching language names for the word 'creole' in Glottologue (`https://glottolog.org/`, last accessed on 30-03-2020), where of the 35 results 14 include English, 8 French in their name. Of course, Spanish, Portuguese, Dutch, Russian, Arabic, Hindi and others also have creoles based on them.

[4] ethnologue.com

them even though the second is a little less similar) may crucially differ.

3.2. Two Scenarios

We describe two LRL scenarios. One with the target language of Nogai[5], a Turkic language of Russia and one with Māori[6], an Austronesian language of New Zealand. The choice of these languages was determined by several factors, a) their alphabets contain few to no special characters which make them difficult scenarios for our classifier, b) they exhibit a number of differences such as alphabet, location, primary lingua franca, language family, typological profile etc. and finally they were accessible to us through previous work, (Hoenen et al., 2020). We defined distractors according to the above logic and criteria.

Language	**Distractors(Type)**, C=Contact, R=Related
Nogai	Russian(C), Kumyk(R), Bashkir(R), Karachai(R), Kazakh(R)
Māori	English(C), Indonesian(R), Tahitian(R), Tongan(R), Samoan(R)

Table 1: The similar languages chosen as distractors for two unrelated LRLs

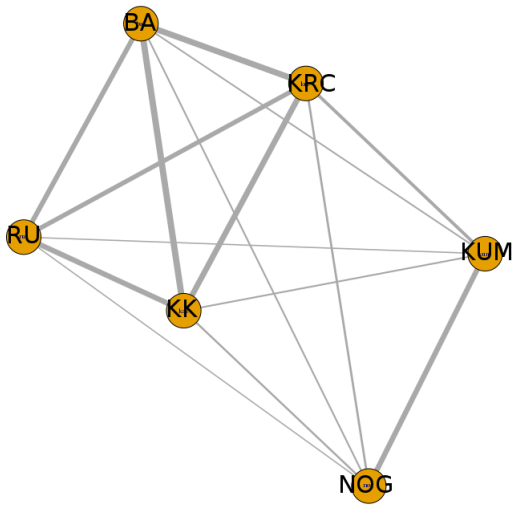

Figure 1: Fully connected graph with token similarities.

Table 1 shows the target languages alongside their distractors. We computed the lexical overlap of the top $10,000$ most frequent tokens in the Nogai subcorpora and produced some visualizations from the concurrent similarity matrix and will briefly discuss them to give a deeper exemplary insight into one of our corpora.

The matrix can be rendered as a fully connected graph, see Figure 1, produced with the R library *igraph*.[7] We see some overlap between Russian, Karachai, Bashkir and Kazakh, while also Nogai and Kumyk share some items. Both Kumyk and Nogai are spoken very close to each other but we did not go furher into this since random patterns may arise in such corpora naturally. An important question is if the data is very noisy. One could imagine that the pattern observed is due to longer sections of Russian in the other languages. However, by means of a language identification heuristic, we had tried to remove longer Russian sections (not single loans) before computing similarity.

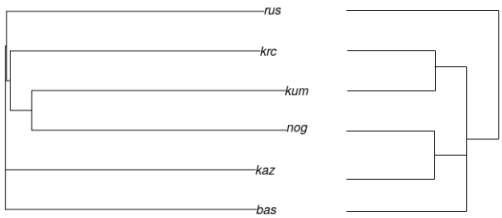

Figure 2: An unrooted neighbor joining tree from the corpus word list similarity data (left) and the genealogy according to Glottologue (right).

Using the Neighbour Joining (Saitou and Nei, 1987) implementation in the R library *phangorn*[8], we generated the unrooted tree in Figure 2 to see if genealogy is instead reflected, which it was not. We display genealogy as in Glottologue[9] building on linguistic sources and considered also (Johanson and Csató, 2015) and (Dryer and Haspelmath, 2013). Indeed, this tree is interesting in respect to its relation to both genealogy and the composition of the respective alphabets.

In order to approach both the shape of this tree and the amount of noise from Russian in the other languages, apart from thorough manual inspection[10], we intersected each language in the corpus with the $20,000$ most frequent words from the Russian National Corpus[11]. We found that the percentages of frequent Russian words was in all cases fairly low. Bashkir and Kazakh, which interestingly also have the alphabets most different from standard Russian,

[5] http://olac.ldc.upenn.edu/language/nog
[6] http://olac.ldc.upenn.edu/language/mri
[7] https://cran.r-project.org/web/packages/igraph/index.html
[8] https://cran.r-project.org/web/packages/phangorn/index.html
[9] https://glottolog.org/resource/languoid/id/noga1249, accessed on 30-03-2020.
[10] Partly, we used machine translation (MT), where we pasted whole subcorpus sections of non-Russian text into online MT APIs such as *DeepL* and *Google Translate* for the automatic translation of Russian to English spotting where the English translation was readable. For the other languages, purity was checked linguistically.
[11] http://ruscorpora.ru/new/ via https://en.wiktionary.org/wiki/Appendix:Frequency_dictionary_of_the_modern_Russian_language_(the_Russian_National_Corpus), accessed on 30-03-2020

adding most letters, showed the lowest rates of overlap with 1.7 % for Bashkir and 2.7% for Kazakh. Karachai had 4.2%, Nogai 4.3% and Kumyk showed the largest overlap with 5.9%. The amounts of frequent Russian words for the group of Nogai, Kumyk and Karachai seem to correlate with the tree. If Kazakh and Bashkir in comparison to the other languages are less likely to loan Russian words unaltered because of their enhanced alphabets this could further explain some of the data. However, the amounts of most frequent Russian words and the similarity were rather incongruent, Russian and Kumyk for instance shared least of their top $10,000$ most frequent words in the corpus, for Kumyk, Karachai shared second most tokens. In summary, whilst being quite pure, the across similarity patterns seem to be influenced by some random factors such as alphabet composition, areal contact and others more than by genealogy. In the other corpus the similarity data reflected genealogy to a much larger degree.

3.3. Writing System

A distractor is only a formidable distractor if it uses the same writing system for otherwise the discrimination can be achieved on first sight without any technical aids. Transliterations especially into the Latin alphabet exist but are generally less standardized than the main writing system. Some languages use more than one writing system. We suggest to split LI into subtasks each concerned with a single writing system if necessary since comparisons across writing systems, for instance of ngrams make little sense. Thus, some distractors can be dismissed immediately.[12]

More often than not, languages use special letters or diacritics or letter combinations in their writing systems (orthographies) distinguishing them even from closely related languages. The case that a pair of languages has a 100% congruent grapheme inventory is rare and an exception rather than the rule. Furthermore, the combination of characters forms a highly significant set. To this end, Wikipedia features a page[13] with a sample of different languages summarizing their use of special characters. Although the information is partly inconcrete, looking at the subset of roughly 50 languages there, which use the Latin alphabet, only 6 of them use only the basic 26 letters, furthermore, Danish with Norwegian and Croatian with Bosnian and Serbian use the same extension. This entails that we have 34 pairs which are indistinguishable qua writing system, thereof only 10 are probable to be present in distractor scenarios (for instance Malay-Indonesian but not Zulu-Norwegian or Zulu-Latin). Taking all possible pairs for 50 languages, we have 2450 possible language pairs and only 10 are possibly underinformative for a classifier, which corresponds to 0.4 percent. There is good reason to believe that the general statistical lesson holds also for other LRLs.

We thus extracted the information on each of the target languages writing systems in Table 1 from Wikipedia which hosts very accurate accounts which we verified and intersected those sets for each pair *target language-distractor language* in our datasets. We use the so obtained sets of exclusive letters later as simple features for classification. Finding a description of the writing system a language uses is much simpler for most LRLs than compiling a corpus of labeled data in order to train statistical LI. Generating pairwise lists allows for maximal information since a more global intersection would result in much fewer features. Likewise, extracting from the to be classified texts in the test set all used letters may be misleading since foreign named entities especially in the contact language may accidentally enhance the document letter sets at hand. Extracting letter sets from training data only may therefore not be able to distinguish between the linguistic core graphemes of a writing system and sporadically occuring foreign characters. Frequency alone may be very low for some core characters such as <x> in many languages which use the Latin alphabet.

3.4. Toponyms

Since, even if this is an exception, some grapheme inventories of writing systems overlap entirely, we use a second source of information both slightly more complex and slighlty less straightforward. The basic idea is that more often than not, a place name is mentioned by texts (in documents) in the main language of that exact place. Especially smaller towns and villages might not be talked of in other languages. Thus the presence is a strong positive hint for LI while the absence does not help to conclude anything. Here, however, more subtleties have to be taken into account. Factors which can influence how probably the mention of a certain place indicates a certain language are:

- the international renown of a place (government, pilgrimage, war, ...)
- mixed populations and languages (in towns languages could be more homogeneous)
- patchwork pattern of different language settlements
- place names which occur multiple times in the world
- etc.

The factor of population size of a settlement subtly plays into many of those factors but by itself may or may not be a priori decisive as to whether a place name is a good candidate. These factors will thus be analyzed as to their occurrence in the data of the target languages and their distractors.

Note that often local toponyms have a different spelling or name in the local and the dominant language (sometimes similar, sometimes calques, sometimes entirely different; for instance Christchuch in Māori is Ōtautahi, an entirely different lexeme) or generally in other languages (compare Venice, originally Venezia, Venedig in German, Venise in French ...). This should, of course, be taken into account compiling language specific toponym lists.

Bootstrapping lists of toponyms is relatively straightforward and we used two different strategies, the first one being the use of place name lists from Wikipedia for Māori,

[12] In fact, some languages as the LRL Yi in China have an exclusive or almost exclusive writing system, where LI simply can use the Unicode Code Block information.

[13] `https://en.wikipedia.org/wiki/Wikipedia:Language_recognition_chart`: last accessed on 18.11.2019

the second the Google Maps Crawler BotSol[14] which allows to extract place names in a variety of languages from manually assigned polygons for Nogai.

3.5. Writing a Classifier

The classifier is a binary classifier. We intend to use it in connection with WIR, which is why the contact languages are especially important.[15] Our prospective task is the build up of an LRL corpus from Web Resources supposing that results of automatic tools such as BootCat (Baroni and Bernardini, 2004) if input were available have to be post-processed from noise through LI. When intending to build a corpus on language X, we are not interested in whether a document which is not in language X is in language Y or Z, hence the set-up as a sequence of binary scenarios is sufficient. For each pair *target-distractor*, we collect lists of

- the exclusive letters (one file per language). If $L1 = \{a, b, c, ...\}; L2 = \{a, b, c, ..\}$, exclusive lettersets are for instance the set of all letters l_i where $i = 1..|L1|$, where $l_i \in L1$ and $l_i \notin L2$
- some letter combinations as mentioned as characteristic according to the sources
- a list of toponyms extracted as described above; for the large contact languages we leave this list empty

For each text in our testsets, we classify the text for each binary scenario simply counting the sum of occurrences (points) of each of the exclusive letters ($P_{L1}(l)$), of the exclusive letter combinations ($P_{L1}(c)$) and of the typical toponyms ($P_{L1}(t)$) per language as an independent language indicator LI_{L1} and LI_{L2}, $LI_{LX} = P_{LX}(l) + P_{LX}(c) + P_{LX}(t)$. We output a decision based on the number of points as probability (here for L1):

$$p(D = L1) = \frac{LI_{L1}}{LI_{L1} + LI_{L2}} \quad (1)$$

. We chain all binary classification scenarios and use a simple majority threshold for the decision of whether to include our document into the corpus or not. We call our classifier LCT-maj (letters, combinations, toponyms - majority vote). So for instance a document D from the testset will be classified as 5 times binarily: Nogai/Kumyk, Nogai/Karachai, Nogai/Russian, Nogai/Kazakh, Nogai/Bashkir so as to end with a vote vector (nog, krc, nog, nog, ba). If there is a nog-majority in the vote vector, we accept the document.
Since our classifier is a binary one, for each pair *target language - distractor*, a number of files (6) have to be produced. If we assume the maximum realistic number of distractors to range between 1 and 11, maximally around 60 files are needed. Whilst this seems a lot, the target language place name file is redundant. In fact, with 11 languages (10 binary pairs) this reduces the number of needed files to actually 51 or $4(n-1)+n$. Whilst this still seems a lot, many of them can and should be produced automatically. Each language pair needs as input a) two files of exclusive letters, b) two files of exclusive letter combinations[16] and c) two files of toponyms. Exclusive letters are those which occur in a languages core grapheme inventory, but which do not occur in the other of the two languages' core grapheme inventory. Given one file with one letter per line for the core grapheme inventories of all languages in the corpus, it is very straightforward to write a small programm to produce all of those files. For the toponyms, we have outlined the use of BotSol above. Producing the files of letter combinations may require some n-Gram extraction or linguistically curated resources. The classifier works also if files are empty out of necessity or lack of information.

3.6. Corpora, Testsets

Language	Number of Tokens	Number of Sites	Size of Wordlist	Size of Placelist
Nogai	57,477	3	15,321	92
Russian	794,603	1	125,509	0
Kumyk	68,347	2	17,191	20
Bashkir	877,827	1	70,116	19
Karachai	269,651	1	49,251	18
Kazakh	973,927	1	130,574	20
Māori	473,375	1	16,882	1,858
English	1,170,472	1	55,373	0
Indonesian	805,072	1	59,250	94
Hawaiian	352,003	1	17,599	146
Tahitian	22,253	1	2,965	18
Tongan	101,847	1	10,265	423
Samoan	129,317	1	12,628	185

Table 2: The corpora and some characteristics. We used a simple space tokenizer first splitting off the usual punctuation marks. Source was most often the Wikipedia.

Table 2 summarizes the two corpora we provide in the same format as the An Crúbadán project (for copyright reasons) albeit adding our place name lists. This includes also source URL information. Additionally, we make our classifier as executable jar available in a generic version and provide the exclusive letter and letter combination lists we used for classification in the binary scenarios.[17] Our two corpora are corpora manually devised from web sources, where as many Wikipedias were extracted as possible by using the tool WikiExtractor[18]. In case of the large languages, we used only the initial section of the Wikipedia (Russian, Indonesian) whereas in English we used the Brown corpus' (Francis and Kucera, 1979) text content (without tags). For Nogai, no Wikipedia was available, so the corpus is manually devised.
The corpora are diverse in terms of size and text types which could be a certain challenge for the training of statistical approaches. For comparison, we used two supervised state-of-the-art tools. The first is langID, (Lui and Baldwin, 2012) which classifies through n-gram statistics and comes with a pretrained model currently recognizing 97 languages. The second is a language identification tool re-

[14] `http://www.botsol.com/Products/GoogleMapsCrawler`, 18.11.2019

[15] They are usually larger (e.g. the governing states main language or the language of former colonial administration) and their content will appear in mixed documents and as results to queries designed to retrieve content only in our target languages. There is more content in these languages on the web.

[16] Inspiring the current approach, for distinguishing Irish and Scottish Gaelic this Youtube user describes an approach using under more diacritics and letter combinations `https://www.youtube.com/watch?v=adg5Ds_9zCA`.

[17] `https://github.com/ArminHoenen/URLCoFi`

[18] `https://github.com/attardi/wikiextractor`

leased through the fasttext website[19] featuring a pretrained model with 176 languages, it operates with embedding vectors internally, see also (Joulin et al., 2016). Both technologies allow to train an own model which we did separately for the two above described corpora. In addition to the corpora, we manually composed testsets with a larger number of documents in the target language and one document per distractor.

We classified each of the documents in the test sets with

- LCT-maj
- langID with the large pretrained model and langID with a model trained on our corpora
- fastText with the large pretrained model and fastText with a model trained on our corpora

Additionally, we used only toponyms for classification and intersected all toponyms with all corpora (of one scenario) in order to see how exclusively the places occurred.

4. Results

Classifier	Accuracy	Failures	Comments
LCT-maj Nogai	$\frac{18}{18}$	-	binary, no non nogai doc had more than $\frac{2}{5}$ nog votes, all nogai docs solely nog votes
fastText (pretrained)	$\frac{3}{18}$	15	model aware of Russian (ru), Kazakh (kk), Bashkir (ba) and Karachai (krc); ru, kk & ba correct krc → ru, Nogai as Russian or Kirghiz
fastText own model	$\frac{18}{18}$	-	
langID (pretrained)	$\frac{2}{18}$	16	model aware of Russian, Kazakh ru, kk correct, Nogai mostly as Russian
langID own model	$\frac{18}{18}$	-	
LCT-maj Māori	$\frac{33}{34}$	1	binary, the rejected document (Māori) was short, loanwords and urls lead to a 3:3 vote
fastText (pretrained)	$\frac{2}{34}$	32	model aware of English and Indonesian no systematic confusion, often English, also Latvian, Welsh, Portuguese etc.
fastText own model	$\frac{34}{34}$	-	
langID (pretrained)	$\frac{2}{34}$	32	model aware of English and Indonesian rather systematic confusion of Māori with Swedish
langID own model	$\frac{32}{34}$	2	same problematic document as in LCT-maj as well as one other Māori → Tahitian

Table 3: Classification of independent test set, which consisted of 18 documents for Nogai, 34 in Māori, one in each distractor. Size chosen for interpretability and availability.

One can see in Table 3 is that the considerably differently composed pretrained models (97 languages for langID, 176 for fastText), which are aware of only a small subset of the required languages in both our scenarios are not useful in our context. But, both individually trained classifiers are extremely accurate and the pronounced differences in sizes of the training corpora do not affect performance of either in our scenarios. The performance of LCT-maj is also on par. Looking into those documents which have been partly classified as another language by either of the classifiers, we found them to contain code-switching or (large) amounts of noise. Thus, rejecting them will lead to a cleaner corpus.

[19] https://fasttext.cc/blog/2017/10/02/blog-post.html

5. Discussion

The results show, that with a very simple input (grapheme inventories, toponyms) which could be bootstrapped relatively easily in our cases, we achieved a statisfactory solution to target language identification for WIR for our LRLs which can compete with state-of-the-art supervised techniques albeit only solving the binary question whether or not a given document is written in a certain LRL and not which other language it is written in. We suspect that this method is applicable to most other smaller languages and especially in WIR for LRL where ressources may be so scarce that acquiring enough training data for any statistical LI approach may be impossible. Furthermore the option might be the only one if part of a Web Corpus Retrieval Pipeline which starts from zero. For larger languages, especially English, the procedure is not applicable as is since there are other languages using the exact same set of letters; also the larger settlement area with placenames appearing multiple times would require different strategies. Thus, the method presented here is primarily one for very small languages with restricted settlement areas and the more idiosyncratic the writing system, the better this is for the method.

Looking into some of the classification results, we note that the vote vectors are often uniform (all votes for Nogai in a Nogai document) for the target language and fully heterogeneous for distractor documents. For some comparisons, the alphabets were very similar and thus uninformative making the system rely more on toponyms. Toponym lists were slightly imbalanced in size (sometimes inevitably so due to the difference in size of the surface area of the settlement areas of the speakers of a language pair). In all, the difference between characters and toponyms as features of a classification helped the system gain robustness.

As to the place names, which were overwhelmingly city and village names, we analyzed them a posteriori and identified those which had most often lead to a misclassification. In all, using only place names was not informative for some documents which simply lacked them (22% of documents for Nogai, 32% for Māori) but classified the others largely correctly for Nogai and Māori. It lead to one false positive in Māori, because the document contained a capitalized noun at a sentence start (where it is indistinguishable from a named entity) which accidentally matched a Māori place. In Nogai, 3 Nogai documents were falsely rejected and 1 Karachai document was identified as Nogai because Nogai places had been talked about. In both cases, the majority of documents was classified correctly. However, here more research is necessary before being able to claim generalizability. To this end, we looked at all mismatched places and ordered them for frequency of mismatch and mismatch (vs. match) ratio (mistmatches divided by matches plus mismatches of that place across binary scenarios).

Table 4 shows some example places and reasons for their mismatches. We looked at characteristics of those places such as popularity, population etc. The goal was to define general rules which can be applied a priori to a toponym list in order to exclude items which can deteriorate performance. We found that

Place	Mismatch	Count	Mismatch Types	Reason
Maitai	1	5	MR → th (5)	homograph in Tahitian
Kihikihi	1	2	MR → tg (2)	homograph in Tongan
Puhi	0.99	474	mr(473), HW(3), tg(1)	Māori place name part
Kereta	0.83	10	MR(2), id(10)	homograph in Indonesian
Stawropol	0.67	33	NOG → ba, kum, krc	administrative seed
Kisliar	1	20	NOG → ba, kum, krc	media coverage

Table 4: Examples of mismatched cities (the city was a cue in the list for the capitalized language but appeared in another corpus). Russian transliterated.

- size
- government/administrative seeds
- accidental (more) frequent lexical homograph in distractor language
- famous places, pilgrimages
- places inhabited by more than one of the concurrent language communities
- place names which are not unique

were such deteriorating factors and some can be excluded a priori.

For the Māori scenario with many languages that have a relatively simple syllable structure and phoneme inventory, the problem with accidental homographs was more pronounced than for Nogai. This suggests a better performance of places for languages with more complex syllable structures and phoneme inventories. English names which appeared as place names in Māori were constantly confused.

Of course for different scenarios weighting to the terms, letters and letter combinations could be introduced depending on such factors. Summarizing, despite the very heterogenous sizes of place name lists, the overall results were good for both scenarios.

6. Conclusions

Recently neural architectures have become popular. In one of the DSL tasks however they performed rather poorly in comparison to other techniques (Malmasi et al., 2016). This shows that statistically sophisticated architectures often but not always represent the most successful approaches. Our approach here of course could be interpreted or implemented in a statistical way using unigram frequencies and highly feature selected tokens (toponyms). But, the binary set-up identifying divergences between grapheme inventories of writing systems would not be equally captured since unigrams would include special characters which occur out of the rule (for instance in loanwords) in the training or test documents. Furthermore, a feature selection scheme ending up with only toponyms (and filtered ones), would be hard to construct. We thus believe that our approach involving hand-picked features, maybe because the task at hand is -compared to others in NLP- relatively simple (even in the face of similar languages) is reasonable even apart from the advantage of using relatively few input data. Furthermore, we believe that our two chosen scenarios are to be interpreted in a hermeneutical way. In hermeneutics sometimes one single example is enough to dismiss the validity of a certain hypothesis. Looking at writing systems in the world, we find languages which have exclusive systems such as Yi. Other writing systems are used for comparatively few languages such as the Georgian letters. Within writing systems, languages (often upon introduction of script) maintain their own letters (often for phonemes not shared with the writing system donor) or in comparison to all other languages using the same writing system a unique combination of letters (or diacritics). Comparing languages only to the set of their distractors further increases uniqueness. This makes it plausible that what we have shown for our two examples extends to many more smaller languages.

We have demonstrated a simple classifier for LI in WIR for LRLs based on writing systems and toponyms. The classifier can compete with state-of-the-art supervised technologies in our case study. It is applicable for scenarios where no labelled data is available (not statistically supervised as it draws only from linguistic descriptions such as graphematic system descriptions and toponyms), but answers only to the binary question whether a document is or is not written in a respective LRL. We provide two corpora including testsets and a customizable binary LI tool for WIR for LRLs. Also, we confirmed the positive capacity of toponyms for LI and identified some rules for a priori exclusion of certain toponyms so as to increase their effect.

7. Acknowledgements

We would like to thank studiumdigitale, the central eLearning authority at Goethe University Frankfurt for their financial support and our reviewers who significantly contributed to improving the content of the paper.

8. Bibliographical References

Barfield, T. (1998). *The Dictionary of Anthropology*. Wiley.

Baroni, M. and Bernardini, S. (2004). Bootcat: Bootstrapping corpora and terms from the web. In *LREC*, page 1313.

Biemann, C., Heyer, G., Quasthoff, U., and Richter, M. (2007). The leipzig corpora collection-monolingual corpora of standard size. *Proceedings of Corpus Linguistic*, 2007.

Chen, S. F. and Maison, B. (2003). Using place name data to train language identification models. In *Eighth European Conference on Speech Communication and Technology*.

Cysouw, M. (2013). Disentangling geography from genealogy. In Peter Auer, et al., editors, *Space in Language and Linguistics: Geographical, Interactional, and Cognitive Perspectives*, pages 21–37, Berlin. De Gruyter Mouton.

Matthew S. Dryer et al., editors. (2013). *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Viviane Déprez et al., editors. (2018). *Negation and Negative Concord: The view from Creoles*. John Benjamins.

Giguet, E. (1995). Categorization according to language: A step toward combining linguistic knowledge and

statistic learning. In *Proceedings of the 4th International Workshop on Parsing Technologies (IWPT-1995), Prague, Czech Republic*.

Hanif, F., Latif, F., and Khiyal, M. S. H. (2007). Unicode aided language identification across multiple scripts and heterogeneous data. *Information Technology Journal*, 6(4):534–540.

Hasimu, M. and Silamu, W. (2017). Three-stage short text language identification algorithm. *Journal of Digital Information Management*, 15(6):354–371.

Henrich, P. (1989). Language identification for the automatic grapheme-to-phoneme conversion of foreign words in a german text-to-speech system. In *First European Conference on Speech Communication and Technology*.

Hoenen, A., Koc, C., and Rahn, M. (2020). A manual for web corpus crawling of low resource languages. *Umanistica Digitale*. forthcoming.

Jauhiainen, T. S., Lui, M., Zampieri, M., Baldwin, T., and Lindén, K. (2019). Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782.

Johanson, L. and Csató, É. (2015). *The Turkic Languages*. Routledge.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification. *CoRR*, abs/1607.01759.

Katz, L. and Frost, R. (1992). The reading process is different for different orthographies: The orthographic depth hypothesis. *Haskins Laboratories Status Report on Speech Research*, SR-111:147–160.

Kilgarriff, A. and Grefenstette, G. (2001). Web as corpus. In *Proceedings of Corpus Linguistics 2001*, pages 342–344. Corpus Linguistics. Readings in a Widening Discipline.

Ljubesic, N., Mikelic, N., and Boras, D. (2007). Language indentification: How to distinguish similar languages? In *2007 29th International Conference on Information Technology Interfaces*, pages 541–546. IEEE.

Lui, M. and Baldwin, T. (2012). langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea, July. Association for Computational Linguistics.

Malmasi, S., Zampieri, M., Ljubešić, N., Nakov, P., Ali, A., and Tiedemann, J. (2016). Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 1–14, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Martins, B. and Silva, M. J. (2005). Language identification in web pages. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 764–768. ACM.

Padró, M. and Padró, L. (2004). Comparing methods for language identification. *Procesamiento del lenguaje natural*, 33.

Ranaivo-Malançon. (2006). Automatic identification of close languages – case study: Malay and indonesian. *ECTI Transactions on Computer and Information Technology*, 2(2):126–134.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.

Samih, Y. and Kallmeyer, L. (2017). *Dialectal Arabic processing Using Deep Learning*. Ph.D. thesis, Ph. D. thesis, Düsseldorf, Germany.

Scannell, K. P. (2007). The crúbadán project: Corpus building for under-resourced languages. In *Building and Exploring Web Corpora: Proceedings of the 3rd Web as Corpus Workshop*, volume 4, pages 5–15.

Thomason, S. G. (2001). *Language Contact*. Georgetown University Press.

Tiedemann, J. and Ljubešić, N. (2012). Efficient discrimination between closely related languages. In *Proceedings of COLING 2012*, pages 2619–2634.

Zampieri, M., Tan, L., Ljubešić, N., and Tiedemann, J. (2014). A report on the DSL shared task 2014. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 58–67, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Zampieri, M., Malmasi, S., Nakov, P., Ali, A., Shon, S., Glass, J., Scherrer, Y., Samardžić, T., Ljubešić, N., Tiedemann, J., et al. (2018). Language identification and morphosyntactic tagging. the second vardial evaluation campaign.

9. Language Resource References

Francis, W. N. and Kucera, H. (1979). *Brown Corpus*. Department of Linguistics, Brown University, Providence, Rhode Island, US.