

Short Text Language Identification for Under Resourced Languages

Bernardt Duvenhage
 Feersum Engine
 Praekelt Consulting
 Johannesburg, South Africa
 bernardt@praekelt.com

Abstract

The paper presents a hierarchical naive Bayesian and lexicon based classifier for short text language identification (LID) useful for under resourced languages. The algorithm is evaluated on short pieces of text for the 11 official South African languages some of which are similar languages.

The algorithm is compared to recent approaches using test sets from previous works on South African languages as well as the Discriminating between Similar Languages (DSL) shared tasks' datasets. Remaining research opportunities and pressing concerns in evaluating and comparing LID approaches are also discussed.

1 Introduction

Accurate language identification (LID) is the first step in many natural language processing and machine comprehension pipelines. If the language of a piece of text is known then the appropriate downstream models like parts of speech taggers and language models can be applied as required.

LID is further also an important step in harvesting scarce language resources. Harvested data can be used to bootstrap more accurate LID models and in doing so continually improve the quality of the harvested data. Availability of data is still one of the big roadblocks for applying data driven approaches like supervised machine learning in developing countries.

Having 11 official languages of South Africa has lead to initiatives (discussed in the next section) that have had positive effect on the availability of language resources for research. However, many of the South African languages are still under resourced from the point of view of building data driven models for machine comprehension and process automation.

Table 1 shows the percentages of first language speakers for each of the official languages of South Africa. These are four conjunctively written Nguni languages (zul, xho, nbl, ssw), Afrikaans (afr) and English (eng), three disjunctively written Sotho languages (nso, sot, tsn), as well as tshiVenda (ven) and Xitsonga (tso). The Nguni languages are similar to each other and harder to distinguish. The same is true of the Sotho languages.

This paper presents a hierarchical naive Bayesian and lexicon based classifier for LID of short pieces of text of 15-20 characters long. The algorithm is evaluated against recent approaches using existing test sets from previous works on South African languages as well as the Discriminating between Similar Languages (DSL) 2015 and 2017 shared tasks.¹

Section 2 reviews existing works on the topic and summarises the remaining research problems. Section 3 of the paper discusses the proposed algorithm and Section 4 presents comparative results.

¹Code and datasets available at <https://github.com/praeekelt/feersum-lid-shared-task>.

Table 1: Percentage of South Africans by First Language

Language (ISO 639)	Percentage	Language (ISO 639)	Percentage
IsiZulu (zul)	23%	isiNdebele (nbl)	2.1%
IsiXhosa (xho)	16%	siSwati (ssw)	2.5%
Afrikaans (afr)	14%	English (eng)	9.6%
Sepedi (nso)	9.1%	Setswana (tsn)	8.0%
Sesotho (sot)	7.6%		
Xitsonga (tso)	4.5%	Tshivenda (ven)	2.4%

2 Related Works

The focus of this section is on recently published datasets and LID research applicable to the South African context. An in depth survey of algorithms, features, datasets, shared tasks and evaluation methods may be found in [1].

The datasets for the DSL 2015 & DSL 2017 shared tasks [2] are often used in LID benchmarks and also available on Kaggle ². The DSL datasets, like other LID datasets, consists of text sentences labelled by language. The 2017 dataset, for example, contains 14 languages over 6 language groups with 18000 training samples and 1000 testing samples per language.

The recently published JW300 parallel corpus [3] covers over 300 languages with around 100 thousand parallel sentences per language pair on average. In South Africa, a multilingual corpus of academic texts produced by university students with different mother tongues is being developed [4]. The WiLI-2018 benchmark dataset [5] for monolingual written natural language identification includes around 1000 paragraphs of 235 languages. A possibly useful link can also be made [6] between Native Language Identification (NLI) (determining the native language of the author of a text) and Language Variety Identification (LVI) (classification of different varieties of a single language) which opens up more datasets. The Leipzig Corpora Collection [7], the Universal Declaration of Human Rights ³ and Tatoeba ⁴ are also often used sources of data.

The NCHLT text corpora [8] is likely a good starting point for a shared LID task dataset for the South African languages [9]. The NCHLT text corpora contains enough data to have 3500 training samples and 600 testing samples of 300+ character sentences per language. Researchers have recently started applying existing algorithms for tasks like neural machine translation in earnest to such South African language datasets [10].

Existing NLP datasets, models and services [11] are available for South African languages. These include an LID algorithm [12] that uses a character level n-gram language model. Multiple papers have shown that 'shallow' naive Bayes classifiers [13, 9, 14, 15], SVMs [16] and similar models work very well for doing LID. The DSL 2017 paper [2], for example, gives an overview of the solutions of all of the teams that competed on the shared task and the winning approach [17] used an SVM with character n-gram, parts of speech tag features and some other engineered features. The winning approach for DSL 2015 used an ensemble naive Bayes classifier. The fasttext classifier [18] is perhaps one of the best known efficient 'shallow' text classifiers that have been used for LID ⁵.

Multiple papers have proposed hierarchical stacked classifiers (including lexicons) that would for example first classify a piece of text by language group and then by exact language [19, 20, 9, 1]. Some work has also been done on classifying surnames between Tshivenda, Xitsonga and Sepedi [21]. Additionally, data augmentation [22] and adversarial training [23] approaches are potentially very useful to reduce the requirement for data.

Researchers have investigated deeper LID models like bidirectional recurrent neural networks [24] or ensembles of recurrent neural networks [25]. The latter is reported to achieve 95.12% in the DSL 2015 shared task. In these models text features can include character and word n-grams as

²<https://www.kaggle.com/vardial/dslcc>

³<http://research.ics.aalto.fi/cog/data/udhr/>

⁴<https://tatoeba.org/eng/downloads>

⁵<https://fasttext.cc/blog/2017/10/02/blog-post.html>

well as informative character and word-level features learnt [26] from the training data. The neural methods seem to work well in tasks where more training data is available.

In summary, LID of short texts, informal styles and similar languages remains a difficult problem which is actively being researched. Increased confusion can in general be expected between shorter pieces of text and languages that are more closely related. Shallow methods still seem to work well compared to deeper models for LID. Other remaining research opportunities seem to be data harvesting, building standardised datasets and creating shared tasks for South Africa and Africa. Support for language codes that include more languages seems to be growing and discoverability of research is improving with more survey papers coming out. Paywalls also seem to no longer be a problem; the references used in this paper was either openly published or available as preprint papers.

3 Methodology

The proposed LID algorithm builds on the work in [9] and [27]. We apply a naive Bayesian classifier with character (2, 4 & 6)-grams, word unigram and word bigram features with a hierarchical lexicon based classifier.

The naive Bayesian classifier is trained to predict the specific language label of a piece of text, but used to first classify text as belonging to either the Nguni family, the Sotho family, English, Afrikaans, Xitsonga or Tshivenda. The scikit-learn multinomial naive Bayes classifier is used for the implementation with an alpha smoothing value of 0.01 and hashed text features.

The lexicon based classifier is then used to predict the specific language within a language group. For the South African languages this is done for the Nguni and Sotho groups. If the lexicon prediction of the specific language has high confidence then its result is used as the final label else the naive Bayesian classifier's specific language prediction is used as the final result. The lexicon is built over all the data and therefore includes the vocabulary from both the training and testing sets.

The lexicon based classifier is designed to trade higher precision for lower recall. The proposed implementation is considered confident if the number of words from the winning language is at least one more than the number of words considered to be from the language scored in second place.

The stacked classifier is tested against three public LID implementations [18, 24, 9]. The LID implementation described in [18] is available on GitHub and is trained and tested according to a post⁶ on the fasttext blog. Character (5-6)-gram features with 16 dimensional vectors worked the best. The implementation discussed in [24] is available from <https://github.com/tomkocmi/LanideNN>. Following the instructions for an OSX pip install of an old r0.8⁷ release of TensorFlow, the LanideNN code could be executed in Python 3.7.4. Settings were left at their defaults and a learning rate of 0.001 was used followed by a refinement with learning rate of 0.0001. Only one code modification was applied to return the results from a method that previously just printed to screen. The LID algorithm described in [9] is also available on GitHub.

The stacked classifier is also tested against the results reported for four other algorithms [17, 27, 25, 16]. All the comparisons are done using the NCHLT [8], DSL 2015 [20] and DSL 2017 [2] datasets discussed in Section 2.

4 Results and Analysis

The average classification accuracy results are summarised in Table 2. The accuracies reported are for classifying a piece of text by its specific language label. Classifying text only by language group or family is a much easier task as reported in [9].

Different variations of the proposed classifier were evaluated. A single NB classifier (NB), a stack of two NB classifiers (NB+NB), a stack of a NB classifier and lexicon (NB+Lex) and a lexicon (Lex) by itself. A lexicon with a 50% training token dropout is also listed to show the impact of the lexicon support on the accuracy.

⁶<https://fasttext.cc/blog/2017/10/02/blog-post.html>

⁷https://github.com/tensorflow/tensorflow/blob/r0.8/tensorflow/g3doc/get_started/os_setup.md

Table 2: LID Accuracy Results. The models we executed ourselves are marked with *. The results that are not available from our own tests or the literature are indicated with '—'.

Model	Algorithm	NCHLT	DSL '15	DSL '17
Joulin et al. 2017 [18] *	fasttext	93.30	93.20	88.60
Bestgen 2017 (DSL winner) [17]	SVM	—	—	92.74
Medvedeva et al. 2017 [16]	SVM	—	—	92.54
Malmasi & Dras 2015 (DSL winner) [27]	NB ensemble	—	95.54	—
Mathur et al. 2017 [25]	RNN ensemble	—	95.12	—
Duvenhage et al. 2017 [9] *	NB+Lex	94.59	—	—
Kocmi & Bojar 2017 [24] *	BRNN	67.84	—	—
Naive-Bayes only *	NB	94.36	94.98	91.89
Stacked model (NB) *	NB+NB	94.41	95.23	91.96
Stacked model (lexicon) *	NB+Lex	96.12	99.34	98.70
Stacked model (50% lex dropout) *	NB+Lex	94.90	98.06	96.21
Lexicon only *	Lex	82.88	97.86	93.56
Lexicon only (sans test data) *	Lex	75.39	81.57	69.74

Table 3: LID requests/sec. on NCHLT dataset. The models we executed ourselves are marked with *. For the other models the results that are not available in the literature are indicated with '—'.

Model:	Joulin 2017 [18]	Duvenhage 2017 [9]	Kocmi 2017 [24]	Proposed
Performance:	44k/s	2.3/s	0.75/s	7.4/s

From the results it seems that the DSL 2017 task might be harder than the DSL 2015 and NCHLT tasks. Also, the results for the implementation discussed in [24] might seem low, but the results reported in that paper is generated on longer pieces of text so lower scores on the shorter pieces of text derived from the NCHLT corpora is expected.

The accuracy of the proposed algorithm seems to be dependent on the support of the lexicon. Without a good lexicon a non-stacked naive Bayesian classifier might even perform better.

The execution performance of some of the LID implementations are shown in Table 3. Results were generated on an early 2015 13-inch Retina MacBook Pro with a 2.9 GHz CPU (Turbo Boosted to 3.4 GHz) and 8GB RAM. The C++ implementation in [18] is the fastest. The implementation in [9] makes use of un-hashed feature representations which causes it to be slower than the proposed sklearn implementation. The execution performance of [24] might improve by a factor of five to ten when executed on a GPU.

5 Conclusion

LID of short texts, informal styles and similar languages remains a difficult problem which is actively being researched. The proposed algorithm was evaluated on three existing datasets and compared to the implementations of three public LID implementations as well as to reported results of four other algorithms. It performed well relative to the other methods beating their results. However, the performance is dependent on the support of the lexicon.

We would like to investigate the value of a lexicon in a production system and how to possibly maintain it using self-supervised learning. We are investigating the application of deeper language models some of which have been used in more recent DSL shared tasks. We would also like to investigate data augmentation strategies to reduce the amount of training data that is required.

Further research opportunities include data harvesting, building standardised datasets and shared tasks for South Africa as well as the rest of Africa. In general, the support for language codes that include more languages seems to be growing, discoverability of research is improving and paywalls seem to no longer be a big problem in getting access to published research.

References

- [1] Tommi Sakari Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782, 2019.
- [2] Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. Findings of the VarDial evaluation campaign 2017. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 1–15, Valencia, Spain, April 2017. Association for Computational Linguistics.
- [3] Željko Agić and Ivan Vulić. JW300: A wide-coverage parallel corpus for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy, July 2019. Association for Computational Linguistics.
- [4] Adelia Carstens and Roald Eiselen. Designing a south african multilingual learner corpus of academic texts (SAMuLCAT). *Language Matters*, 50(1):64–83, 2019.
- [5] Martin Thoma. The WiLI benchmark dataset for written language identification, 2018.
- [6] Marc Franco-Salvador, Greg Kondrak, and Paolo Rosso. Bridging the native language and language variety identification tasks. *Procedia computer science*, 112:1554–1561, 2017.
- [7] Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 759–765, Istanbul, Turkey, May 2012. European Languages Resources Association (ELRA).
- [8] NCHLT text corpora, 2014. Available from <http://www.nwu.ac.za/ctext>.
- [9] Bernardt Duvenhage, Mfundo Ntini, and Phala Ramonyai. Improved text language identification for the south african languages. *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pages 214–218, 2017.
- [10] Jade Z Abbott and Laura Martinus. Towards neural machine translation for african languages. *arXiv preprint arXiv:1811.05467*, 2018.
- [11] Martin Puttkammer, Roald Eiselen, Justin Hocking, and Frederik Koen. NLP web services for resource-scarce languages. In *Proceedings of ACL 2018, System Demonstrations*, pages 43–49, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [12] Justin Hocking. Language identification for south african languages. In *Proceedings of the Annual Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech): Poster session.*, 2014.
- [13] Alexandra Espichán-Linares and Arturo Oncevay-Marcos. Language identification with scarce data: A case study from peru. In Juan Antonio Lossio-Ventura and Hugo Alatrasta-Salas, editors, *Information Management and Big Data*, pages 90–105, Cham, 2018. Springer International Publishing.
- [14] Adrien Barbaresi. Discriminating between similar languages using weighted subword features. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, 2017.
- [15] Tommi Jauhiainen, Krister Lindén, and Heidi Jauhiainen. Evaluation of language identification methods using 285 languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, 2017.
- [16] Maria Medvedeva, Martin Kroon, and Barbara Plank. When sparse traditional models outperform dense neural networks: the curious case of discriminating between similar languages. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, 2017.
- [17] Yves Bestgen. Improving the character ngram model for the DSL task with BM25 weighting and less frequently used feature sets. In *Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial)*, pages 115–123, Valencia, Spain, April 2017. Association for Computational Linguistics.

- [18] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain, April 2017. Association for Computational Linguistics.
- [19] Cyril Goutte, Serge Léger, and Marine Carpuat. The NRC system for discriminating similar languages. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 139–145, Dublin, Ireland, August 2014. Association for Computational Linguistics and Dublin City University.
- [20] Marcos Zampieri, Binyam Gebrekidan Gebre, Hernani Costa, and Josef van Genabith. Comparing approaches to the identification of similar languages. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 66–72, Hissar, Bulgaria, September 2015. Association for Computational Linguistics.
- [21] Tshephisho Joseph Sefara, Madimetja Jonas D. Manamela, and Promise Tshepiso Malatji. Text-based language identification for some of the under-resourced languages of south africa. *2016 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, pages 303–307, 2016.
- [22] Vukosi N. Marivate and Tshephisho Sefara. Improving short text classification through global augmentation methods. *CoRR*, abs/1907.03752, 2019.
- [23] Yitong Li, Timothy Baldwin, and Trevor Cohn. What’s in a domain? learning domain-robust text representations using adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 474–479, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
- [24] Tom Kocmi and Ondřej Bojar. LanideNN: Multilingual language identification on text stream. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 927–936, Valencia, Spain, April 2017. Association for Computational Linguistics.
- [25] Priyank Mathur, Arkajyoti Misra, and Emrah Budur. LIDE: Language identification from text documents. *ArXiv*, abs/1701.03682, 2017.
- [26] Aaron Jaech, George Mulcaire, Shobhit Hathi, Mari Ostendorf, and Noah A. Smith. Hierarchical character-word models for language identification. pages 84–93, November 2016.
- [27] Shervin Malmasi and Mark Dras. Language identification using classifier ensembles. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 35–43, Hissar, Bulgaria, September 2015. Association for Computational Linguistics.