Optimizing Naive Bayes for Arabic Dialect Identification

Tommi Jauhiainen, Heidi Jauhiainen, Krister Lindén
Department of Digital Humanities, University of Helsinki, Finland
tommi.jauhiainen@helsinki.fi

Abstract

This article describes the language identification system used by the SUKI team in the 2022 Nuanced Arabic Dialect Identification (NADI) shared task. In addition to the system description, we give some details of the dialect identification experiments we conducted while preparing our submissions. In the end, we submitted only one official run. We used a Naive Bayes-based language identifier with character n-grams from one to four, of which we implemented a new version, which automatically optimizes its parameters. We also experimented with clustering the training data according to different topics. With the macro F1 score of 0.1963 on test set A and 0.1058 on test set B, we achieved the 18th position out of the 19 competing teams.

1 Introduction

This paper describes the system used by the SUKI team at the Nuanced Arabic Dialect Identification (NADI) shared task 2022 (Abdul-Mageed et al., 2022). The task was the third in a series of language identification shared tasks focusing on Arabic languages (Abdul-Mageed et al., 2020, 2021b). In 2020, the first subtask of country-level classification was won by Talafha et al. (2020) using multi-dialect Arabic BERT model (Devlin et al., 2019) and the second subtask of province-level classification by El Mekki et al. (2020) using an ensemble of a BERT-based and a stochastic gradient descent (SGD) based (Zhang, 2004) identifiers. The various subtasks of the 2021 edition were won by AlKhamissi et al. (2021) using MARBERT-based systems (Abdul-Mageed et al., 2021a). A recent literature review of language identification for dialectal Arabic was conducted by Elnagar et al. (2021) and a more general survey of language identification techniques by Jauhiainen et al. (2019d). Deep learning, specifically BERT-based, systems dominated the two previous NADI shared tasks.

As the SUKI team, we have participated in various language identification (LI) related shared tasks throughout the years with our shallow HeLI or Naive Bayes-based systems. In 2016, we participated in the Arabic dialect sub-task of the 3rd edition of the Discriminating Between Similar Languages (DSL) shared task, which featured four Arabic dialects in addition to Modern Standard Arabic (MSA) (Jauhiainen et al., 2016). Using the HeLI LI method, we arrived at the seventh position, which was poor in contrast to the shared first place we reached in the first sub-task of DSL that year. The experiments described in this paper are the first time we have returned to the identification of various Arabic languages after that. In these experiments, we use a Naive Bayes (NB) based identifier instead of one based on the HeLI method. We implemented it and used it as a baseline in the 2019 Cuneiform Language Identification (CLI) shared task (Jauhiainen et al., 2019a). During the same year, we adapted our language model adaptation scheme (Jauhiainen et al., 2019c) to work with the NB implementation and won one of the two tracks in the Discriminating between the Mainland and Taiwan variation of Mandarin Chinese (DMT, Zampieri et al. (2019)) shared task (Jauhiainen et al., 2019b). More recently, we also won the Romanian Dialect Identification (RDI, Chakravarthi et al. (2021)) 2021 (Jauhiainen et al., 2021) and the Identification of Languages and Dialects of Italy (ITDI, Aepli et al. (2022)) 2022 (Jauhiainen et al., 2022a) shared tasks using the adaptive version of the NB identifier.

For the NADI shared task, we set out to find out whether our current NB implementation would be more competitive when distinguishing between close Arabic languages than our HeLI-based identifier in 2016. Additionally, we were trying to develop a way to use unlabeled data to improve the identifier results. The experiments to utilize unlabeled data were inconclusive and did not improve

Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP), pages 409 - 414

the identification results on the development set, so we did not end up using them in the one run we submitted. Also, as the language identification accuracy was already relatively low, using language model adaptation did not prove advantageous with the development data. Thus we submitted our only run using the non-adaptive NB identifier.

2 Shared Task Evaluation Setting

The third NADI shared task[1] featured 18 country-level dialects of Arabic. The official ranking metric was the macro-averaged F1 score. The shared task participants were given separate training and development sets consisting of tweets labeled with their respective country-level dialects. The training set was the same as in the NADI 2021 shared task (Abdul-Mageed et al., 2021b). According to the shared task instructions, the provided development set was not to be used as training data for the identifier used for the test data. The set sizes are seen in Table 1.

The participants were also given the tweet IDs of 10 million additional unlabeled Arabic tweets that could be used in training and developing the language identification system. The organizers provided a Python script that could be used to download the corresponding tweets using a Twitter API and their credentials. Currently, Twitter allows Academic users to download 10 million monthly tweets for research purposes. Due to the Twitter service being repeatedly over capacity and terminating the connection, the download had to be made in 16 parts, which took almost a week. Of the 9,999,998 downloaded tweets, 2,005,682 were tagged as **<UNAVAILABLE>**.

The participants were expected to provide results on two test sets; test set A featuring new unseen tweets for each of the 18 dialects and test set B featuring tweets from a subset of unknown size from the 18 languages.

We only used the NADI-labeled training and development sets for the submitted run. We did not use the development set for training the final identifier; we used it only to determine the method's optimal parameters.

3 System

The system uses a Naive Bayes-based method using the observed relative frequencies of multiple-size character n-grams as probabilities. As described by Jauhiainen et al. (2022a), the Naive Bayes type method adds together logarithms of the relative frequencies of character *n*-gram combinations f_i in the training data C_g as defined in Equation 1:

$$R(g,M) = -lg_{10} \prod_{i=1}^{\ell_{M^F}} v_{C_g}(f_i) = \sum_{i=1}^{\ell_{M^F}} -lg_{10}(v_{C_g}(f_i)) \quad (1)$$

where ℓ_{M^F} is the number of individual features in the mystery text M to be identified, and f_i is M's ith feature. The relative frequency, $v_{C_g}(f)$, is calculated as in Equation 2:

$$v_{C_g}(f) = \begin{cases} \frac{c(C_g,f)}{\ell_{C_g^F}}, \textit{ if } c(C_g,f) > 0 \\ \frac{1}{\ell_{C_g^F}} pm, \textit{ otherwise} \end{cases} \quad (2)$$

where $c(C_g, f)$ is the count of feature f in the training corpus C_g of the language g. $\ell_{C_g^F}$ is the length of the corpus C_g when it has been transformed into a collection of features F, e.g., features of the same type as f. The pm is the penalty modifier, which is optimized using the development data.

The exact range of the used character n-grams is optimized using the development data. In previous versions of the identifier, we have semi-manually identified the optimal character n-gram ranges and the penalty modifier. However, on this occasion, we decided to implement an automatic optimizer to streamline experimentation. The automatic optimizer is first given initial character n-gram and penalty modifier ranges which it then uses to populate a todo-table. The parameters in the todo-table are evaluated, and the results are stored in a master results list. An additional top ten list of macro F1 scores is created with the parameters used to obtain them. The parameter instances used in the top ten list are checked, and nearby parameter combinations are added to a new todo-table if they are not found in the master results list. In the case of n-gram ranges, the optimizer tries one higher and one lower for both the minimum and maximum n-gram sizes. For the penalty modifier, it adds and subtracts 0.5 from the current one if there are no other penalty modifiers for the respective n-gram range in the master results list. If a "neighboring" penalty modifier exists in the results list, the halfway between the penalty modifiers is tried if the distance between modifiers is larger than 0.1. The cycle of evaluating the todo-table, creating a top ten list, and creating a new todo-table is continued as long as the top ten list changes between cycles. An ex-

[1] http://nadi.dlnlp.ai

Country	# tweets train	# tweets dev.	# tweets test A	# tweets test B
Egypt	4,283	1,041	?	?
Iraq	2,729	664	?	?
Saudi Arabia	2,140	520	?	?
Algeria	1,809	430	?	?
Oman	1,501	355	?	?
Syria	1,287	278	?	?
Libya	1,286	314	?	?
Tunisia	859	173	?	?
Morocco	858	207	?	?
Lebanon	644	157	?	?
United Arab Emirates	642	157	?	?
Yemen	429	105	?	?
Kuwait	429	105	?	?
Jordan	429	104	?	?
Palestine	428	104	?	?
Sudan	215	53	?	?
Qatar	215	52	?	?
Bahrain	215	52	?	?
Total	20,398	4,871	4,758	1,474

Table 1: The number of tweets of each Arabic dialect in the training and development sets of the NADI 2022 shared task.

n-gram range	penalty modifier
1 – 4	1.3
2 – 4	1.3
1 – 5	1.5
1 – 5	1.8

Table 2: An example of a master results list for the automatic optimizer.

n-gram range	penalty modifier
1 – 3	1.3
1 – 5	1.3
1 – 4	1.8
1 – 4	0.8
2 – 5	1.3
3 – 4	1.3
2 – 4	0.8
2 – 4	1.8
1 – 6	1.5
1 – 4	1.5
2 – 5	1.5
1 – 5	1.0
1 – 5	1.65
1 – 6	1.8
2 – 5	1.8
1 – 5	2.3

Table 3: An example todo-table generated on basis of master results list in Table 2.

ample of creating a todo-table from a top ten list is given in Tables 2 and 3.

We have published the code of the version used in the NADI shared task on GitHub.[2]

The only external part of our language identification pipeline was the Farasa morphological segmentation tool (Abdelali et al., 2016).[3] It had been

[2] https://github.com/tosaja/TunPRF-NADI

[3] https://farasa.qcri.org/segmentation/

# splits	Macro F1
1	0.2049
2	0.2038
4	0.2011
8	0.2011
16	0.1980

Table 4: The results of the adaptation experiments on the development data.

successfully used in the NADI shared task before by El Mekki et al. (2020) and Wadhawan (2021), and by Alrifai et al. (2017) already in the 5th Author Profiling Task at PAN 2017 (Rangel et al., 2017). When the tweets are run through Farasa, it adds "+" characters between morphemes.

4 Experiments

Manually optimizing the parameters for the NB system, we arrived at the Macro F1 score of 0.2046 with n-grams from two to four and the penalty modifier of 1.40. After this, we did some experiments with language model adaptation using the same parameters, but adding more splits to adaptation worsened the results, as seen in Table 4. There was a slight increase in the F1 score, which indicated that some form of adaptation might be beneficial. However, it was clear that the accuracy of the identifier was too low for adaptation to have any meaningful effect, which is why we decided to leave adaptation experiments until our non-adaptive identification system would produce considerably better results.

The implemented automatic optimizer arrived at the macro F1 score of 0.2070 using character

Macro F1	n-gram range	penalty modifier
0.2119	1 – 4	1.375
0.2111	2 – 4	1.375
0.2106	2 – 5	1.5
0.2104	1 – 5	1.5
0.2094	1 – 5	1.5625
0.2087	1 – 4	1.4375
0.2082	2 – 4	1.3125
0.2078	1 – 5	1.625
0.2077	2 – 5	1.5625
0.2072	1 – 4	1.3125

Table 5: The final top 10 scores with their parameters on the development set. Farasa segmenter was used on both the training and the development data.

# tweets in cluster	# clusters	# lang. combinations	Macro F1
2	61	1,119	0.1733
3	44	1,037	0.1682
4	15	935	0.1632
5	10	891	0.1607
6–9	19	858	0.1597
10–19	25	767	0.1540
20–39	16	588	0.1476
40–99	10	408	0.1378
100–199	4	263	0.1361
200–399	5	197	0.1413
400–999	2	108	0.1550
2,485	1	72	0.1748
3,674	1	54	0.1834
8,933	1	36	0.1964

Table 6: The results of the clustering experiments on the development data. The total number of clusters in the "# clusters" column is 214. The "# lang. combinations" column indicates the total number of the cluster – language combinations after all the clusters on the corresponding row and above were combined into one cluster.

n-grams from one to four with a penalty modifier of 1.4375. The 0.002 score difference, when compared with the manual optimization results, was due to adding a space character at the beginning and the end of each tweet in the training data–a trick we had already done to the tweets being tested. We arrived at slightly better results using the optimizer with the Farasa-treated training and development sets. The top ten combinations with their macro F1 scores after running the automatical optimizer on the Farasa-treated training and development data can be seen in Table 5. We have not used any morphological segmentation with the NB identifier in our previous language identification experiments and cannot say whether using such segmentation is generally advantageous. The observed 2,4% macro F1 score improvement in this dataset could actually be a random effect.

Clustering Experiments Dividing languages into topic- or dialect-based clusters has proven fruitful in our earlier experiments (Jauhiainen et al., 2022b). We expected the training data to contain Tweets on many different topics and hypothesized that dividing the training data into several clusters might be advantageous. Each dialect would then be divided into several language models based on these clusters.

We created a custom clustering software based on the Naive Bayes identifier. It chose a random tweet among all the tweets and created language models from it. Then every other tweet was scored using those language models, and the one furthest from the original tweet was selected. Additional language models were also created from the second tweet, and then again, all the tweets were identified using both models. If the model claimed only one tweet, e.g., itself, the model was dropped out of the repertoire as an outlier. Then the tweet being as far as possible from both models was selected as the material for the third model. And again, all the tweets were re-scored, one chosen for new models, and so on until none of the models claimed more than half of all the tweets (max 10k tweets). This resulted in 214 clusters for all the dialects, as seen in Table 6. The displayed F1 scores are the best results on the development set after all the clusters on the corresponding row and above were combined into one cluster. The results of the clustering experiments were not good enough for the clustering to be used in an actual submission to the shared task. We still had some further ideas of how to try to improve the results but were unable to continue due to limited time.

5 Results

We ended up submitting only one run on each of the test sets using the non-adaptive version of the language identifier. First, we treated both the training and the test data with the Farasa segmenter and then ran them through the Naive Bayes language identifier using character n-grams from one to four with a penalty modifier of 1.375. With the macro F1 score of 0.1963 on test set A and 0.1058 on test set B, our submissions reached the 19/19 and 15/19 positions for the respective test sets. The final ranking for the whole shared task combined the results of the two test sets. We were ranked 18th out of the 19 participating teams, which shows that our results were not competitive against most other

submitted results. As of this writing, we have not received the gold-standard labels for the test set.

6 Discussion

There are still several avenues worth exploring when using the NB-based identifier in classifying Arabic tweets. We intend to continue exploring different kinds of topic clustering methods to divide the training data into different models. Currently, we have no efficient means to utilize additional unannotated data, and developing such means remains a high priority.

7 Conclusion

We have presented the experiments we conducted when participating in the NADI 2022 shared task. Many of the experiments provided interesting results for further research. We were successful in implementing a new version of the NB identifier, which automatically optimizes its parameters, thus leaving more time to explore ideas to improve the identification accuracy. We reached the 19th and 15th places in the shared task.

Limitations

As seen from the results of the shared task, using a shallow NB identifier with character n-grams is not currently competitive against BERT-based deep learning systems in classifying Arabic tweets according to their origin countries. These experiments serve well in pointing out the limitations of a system that has won several other language identification shared tasks (Jauhiainen et al., 2019b, 2021, 2022a).

Acknowledgements

The research was conducted within the Language Identification of Speech and Text project funded by the Finnish Research Impact Foundation from its Tandem Industry Academia funding in cooperation with Lingsoft.

References

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for Arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–16, San Diego, California. Association for Computational Linguistics.

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021a. ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, Houda Bouamor, and Nizar Habash. 2020. NADI 2020: The first nuanced Arabic dialect identification shared task. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 97–110, Barcelona, Spain (Online). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2021b. NADI 2021: The second nuanced Arabic dialect identification shared task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 244–259, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2022. NADI 2022: The Third Nuanced Arabic Dialect Identification Shared Task. In *Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP 2022)*.

Noëmi Aepli, Antonios Anastasopoulos, Adrian Chifu, William Domingues, Fahim Faisal, Mihaela Găman, Radu Tudor Ionescu, and Yves Scherrer. 2022. Findings of the VarDial evaluation campaign 2022. In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, Gyeongju, Republic of Korea.

Badr AlKhamissi, Mohamed Gabr, Muhammad El-Nokrashy, and Khaled Essam. 2021. Adapting MARBERT for improved Arabic dialect identification: Submission to the NADI 2021 shared task. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 260–264, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Khaled Alrifai, Ghaida Rebdawi, and Nada Ghneim. 2017. Arabic Tweeps Gender and Dialect Prediction – Notebook for PAN at CLEF 2017. In *Working Notes Papers of CLEF 2017 Evaluation Labs and Workshop*, Dublin, Ireland. CEUR-WS.org.

Bharathi Raja Chakravarthi, Mihaela Gaman, Radu Tudor Ionescu, Heidi Jauhiainen, Tommi Jauhiainen, Krister Lindén, Nikola Ljubešić, Niko Partanen, Ruba Priyadharshini, Christoph Purschke, Eswari Rajagopal, Yves Scherrer, and Marcos Zampieri. 2021. Findings of the VarDial evaluation campaign 2021. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–11, Kyiv, Ukraine. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Abdellah El Mekki, Ahmed Alami, Hamza Alami, Ahmed Khoumsi, and Ismail Berrada. 2020. Weighted combination of BERT and n-GRAM features for nuanced Arabic dialect identification. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 268–274, Barcelona, Spain (Online). Association for Computational Linguistics.

Ashraf Elnagar, Sane M. Yagi, Ali Bou Nassif, Ismail Shahin, and Said A. Salloum. 2021. Systematic Literature Review of Dialectal Arabic: Identification and Detection. *IEEE Access*, 9:31010–31042.

Tommi Jauhiainen, Heidi Jauhiainen, Tero Alstola, and Krister Lindén. 2019a. Language and Dialect Identification of Cuneiform Texts. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 89–98, Ann Arbor, Michigan. Association for Computational Linguistics.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2019b. Discriminating between Mandarin Chinese and Swiss-German varieties using adaptive language models. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 178–187, Ann Arbor, Michigan. Association for Computational Linguistics.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2021. Naive Bayes-based experiments in Romanian dialect identification. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 76–83, Kyiv, Ukraine. Association for Computational Linguistics.

Tommi Jauhiainen, Heidi Jauhiainen, and Krister Lindén. 2022a. Italian language and dialect identification and regional French variety detection using adaptive naive bayes. In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, Gyeongju, Republic of Korea.

Tommi Jauhiainen, Krister Lindén, and Heidi Jauhiainen. 2016. HeLI, a word-based backoff method for language identification. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 153–162, Osaka, Japan. The COLING 2016 Organizing Committee.

Tommi Jauhiainen, Krister Lindén, and Heidi Jauhiainen. 2019c. Language model adaptation for language and dialect identification of text. *Natural Language Engineering*, 25(5):561–583.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2019d. Automatic Language Identification in Texts: A Survey. *Journal of Artificial Intelligence Research*, 65:675–782.

Tommi Jauhiainen, Jussi Piitulainen, Erik Axelson, and Krister Lindén. 2022b. Language identification as part of the text corpus creation pipeline at the language bank of finland. In *Digital Humanities in Nordic and Baltic Countries conference (DHNB 2022)*.

Francisco Rangel, Paolo Rosso, Martin Potthast, and Benno Stein. 2017. Overview of the 5th Author Profiling Task at PAN 2017: Gender and Language Variety Identification in Twitter. In *Working Notes Papers of CLEF 2017 Evaluation Labs and Workshop*, Dublin, Ireland. CEUR-WS.org.

Bashar Talafha, Mohammad Ali, Muhy Eddin Za'ter, Haitham Seelawi, Ibraheem Tuffaha, Mostafa Samir, Wael Farhan, and Hussein Al-Natsheh. 2020. Multi-dialect Arabic BERT for country-level dialect identification. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 111–118, Barcelona, Spain (Online). Association for Computational Linguistics.

Anshul Wadhawan. 2021. Dialect identification in nuanced Arabic tweets using farasa segmentation and AraBERT. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 291–295, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Yves Scherrer, Tanja Samardžić, Francis Tyers, Miikka Silfverberg, Natalia Klyueva, Tung-Le Pan, Chu-Ren Huang, Radu Tudor Ionescu, Andrei M. Butnaru, and Tommi Jauhiainen. 2019. A report on the third VarDial evaluation campaign. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–16, Ann Arbor, Michigan. Association for Computational Linguistics.

Tong Zhang. 2004. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116.