Is Encoder-Decoder Transformer the Shiny Hammer?

Nat Gillin
Gillin Inc.
36 Natchez Street, Seahaven, USA
gillin.nat@gmail.com

Abstract

We present an approach to multi-class classification using an encoder-decoder transformer model. We trained a network to identify French varieties using the same scripts we use to train an encoder-decoder machine translation model. With some slight modification to the data preparation and inference parameters, we showed that the same tools used for machine translation can be easily re-used to achieve competitive performance for classification. On the French Dialectal Identification (FDI) task, we scored 32.4 on weighted F1, but this is far from a simple naive Bayes classifier that outperforms a neural encoder-decoder model at 41.27 weighted F1.

1 Introduction

Sometimes one might find more appealing to re-use the same code, scripts and infrastructure that already serve an NLP product for another purpose.

In this case, an eco-system of tools is already available to train machine translation models and serve the model with a RESTful API, then we need some language identification tools. Then, one might think,

> *Technically, an auto-regressive encoder-decoder model that produces a single token at inference is sort of like a classifier.*

Recent works had validated the thought (Li et al., 2018; Thant and Nwet, 2020; Hadar and Shmueli, 2021), most notably the "Don't Classify, Translate!" (DCT) idea simply re-used an encoder-decoder machine translation models as a hierarchical classifier to categorize e-commerce products.

To test the DCT model for language identification, we evaluated the approach on the French Cross-Domain Dialect Identification (FDI) dataset (Gaman et al., 2022) while participating in a Vardial shared task .[1]

An example of the input and output of the FDI data looks as follows:

> **[IN]:** *Le NE compte une importante communauté ukrainienne qui s'élève à environ 1,3 million de personnes.*
>
> **[OUT]:** BE

where the input text sometimes contains named-entities and they are masked with the `$NE$` token and the output is a two-char locale code to roughly represent the dialect.

2 Motivation

Our initial thought was to use the least effort in script changes to train a machine translation model to a multi-class classification one. Being frugal, the secondary objective is to ensure that we do not spend more than a day's worth of GPU hours.

Intuitively, we need the decoder to produce only one token that marks the class label, so we shouldn't be needing heavy machinery (i.e. deep layers) in the decoder. Previous works (Domhan et al., 2020; Susanto et al., 2019) have also shown that offsetting decoder layers with more encoder layers could improve inference latency. Also, when training encoder-decoder models on small datasets, deep decoder layers might be an overkill.

Therefore, we decided to re-use a "*mini*" transformer (Vaswani et al., 2017) with 6 encoder, 2 decoder layers trained with the Marian NMT toolkit (Junczys-Dowmunt et al., 2018).[2]

3 TL;DR (Experimental Setup)

We trained an encoder-decoder machine translation model using the Marian NMT framework with the following hyperparameters:

[1] https://sites.google.com/view/vardial-2022/shared-tasks

[2] Using this script from https://github.com/alvations/myth/blob/master/train-sarah.sh

Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects, pages 80–85
October 16, 2022. ©2022 Association for Computational Linguistics

- **Transformer with 6 encoder, 2 decoder,**
  - 8 attention heads
  - vocabulary size of 8,000
  - embedding dimension of 1024
  - transformer feed-forward dim. of 4096
- **Adam optimizer parameters**
  - learning rate sets warm-up at 8,000
  - max learning rate set to 0.0001
  - inverse square root learning rate decay
- **Sentencepiece options**
  - character coverage was set to 100%
  - class labels were set as user-defined symbols, viz. `BE, CA, CH, FR` to represent *Belgian*, *Canadian*, *Swiss* and *France* French varieties.
  - the same sentencepiece vocabulary is used for the source input and target output
- **Data limit options**
  - *during training*, the maximum length of the text input were cropped to 1,000 sentencepieces
  - *during validation*, the maximum length of the text input was set to 5,000 sentencepieces
  - *at inference*, when applying it to the test set, the max length was set to 500 sentencepieces[3]
- **Other notable hyperparameters**
  - global dropout regularization was set at 0.1
  - beam size was set to 3 during inference
  - label backoff when decoder produces output that is not any of the label

The modified script with the above hyperparameter used to train the model is available on https://github.com/alvations/myth/blob/master/train-esther.sh. We refer to this model as `DCT mini` for the rest of the paper.

[3] ~~*Cos*~~ Because we wanted to keep the inference time tractable in production, i.e. <300ms

3.1 How Low Can We Go?

To push the limits of the '*Don't Translate, Classify*' approach, we want to see how the smallest possible model performs on the FDI dataset. We trained a model with transformer with ***1 encoder, 1 decoder and 1 attention head***. The rest of the hyperparameters are same as the ones described Section 3 above. We refer to this model as `DCT micro` for the rest of the paper.

3.2 Non-neural Baseline

Additionally, to compare our models with a non-neural baseline, we trained a naive Bayes model similar to the ones reported in Tan et al. (2014).[4]. Sweeping through 1 to 12 character n-grams features, the best model based validated on the development is based on 6 to 10 character n-grams. We refer to this model as `Naive Bayes` for the rest of the paper.

4 Results

Systems	Micro	Macro	Weighted
`Naive Bayes`	45.82	31.19	41.27
`DCT Mini`	39.14	26.27	32.35
`DCT Micro`	34.21	19.05	24.16
NRC	**49.34**	**34.37**	**45.81**
SUKI	39.18	26.61	34.22

Table 1: F1-scores of the Systems on the FDI Test Set

Table 1 reports the F1-scores of the systems we mentioned earlier and the best systems' results of the other teams (NRC and SUKI) that participated in the shared task (Aepli et al., 2022).

The `Naive Bayes` baseline result is unsurprisingly strong and the DCT approaches were competitive but much weaker at around 10 points F1-score lower. While we expected a drop in quality, the drastic F1 score drop from `DCT Mini` to `DCT Micro` is startling. A naive probabilistic model outperforming neural models on classification task is not a novel finding (Bernier-Colborne et al., 2019) and sometimes neural models when trained inappropriately with bad hyperparameter sets do not outperform the old-school statistical/probabilistic approaches (Nat, 2016; Zhang and Duh, 2020).

[4] Using script from https://github.com/alvations/bayesline-DSL/blob/master/dsl-2019.py

4.1 A Naive Bayesline

We note a performance difference of the naive Bayes models between the validation and test data. In retrospect, evaluating the naive Bayes models on the test data labels, the best feature is 4 to 6 character n-grams, and it achieves the 44.98 weighted F1 score, 34.33 and 47.15 on macro and micro F1 scores. But note that picking the best model based on such oracle knowledge is unrealistic.

The difference between the model selected based on the validation results and the test gold standard reflects possibly a difference in data distribution and Ng (2016) would suggest to collect more validation data so that the difference between the validation and test set is kept to a minimum.

5 Analysis

Figure 1 and 2 presents the confusion matrices for the `DCT mini` and `DCT micro` models.

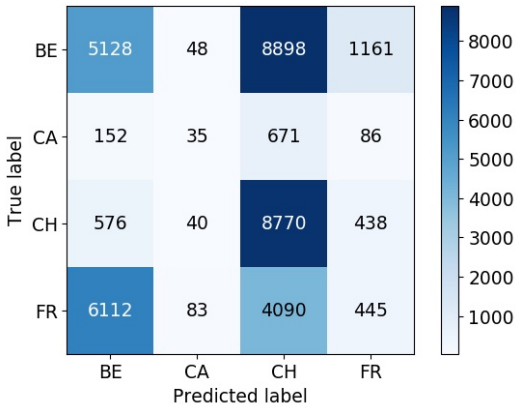

Figure 1: Confusion Matrix for `DCT mini`

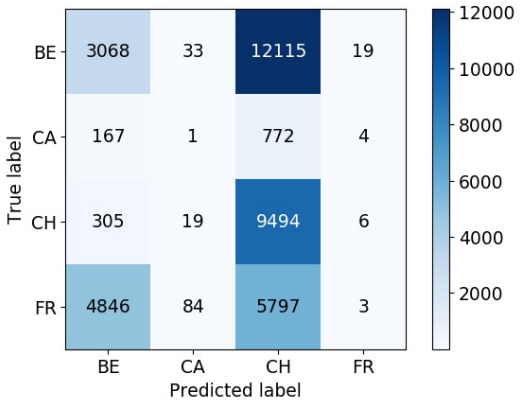

Figure 2: Confusion Matrix for `DCT micro`

For both models, we observe that the:

- `FR` label was commonly misidentified as `BE` or `CH`
- `BE` label was commonly misidentified as `CH`
- true positive rate for the `CA` label is relatively low compared to other labels

Specific to the `DCT mini` model, it has higher false positive rate when wrongly classifying `BE` as `FR` while the `DCT micro` did not present this behavior.

5.1 Label Class Distribution

One possible suspicion for the high false positives on `CH` and `FR` in the test set might be due to the training/validation label distribution. Ideally, a robust language identification should not be affected by the label class distribution of the training and validation data.

But label distribution is not the culprit here, Table 2 gives no evidence of the DCT model biasing label classes that resembles training/validation distribution. This is unlike classical classification models that requires imbalanced data.

	Training	Validation	Test	Predicted
BE	33.93	42.9	41.47	33.26
CA	9.48	0.95	2.57	0.57
CH	39.37	29.13	26.74	62.33
FR	17.22	27.02	29.21	3.85

Table 2: Label Class Distribution of the Training, Validation, Test Data and the Predicted Labels from the `DCT Mini` model.

5.2 The FDI Dataset

If you've read till now, you would have realized that we deliberately avoided in-depth exploratory data analysis before we trained discussed model training and the results. That is because we know that ***there will be issues with any dataset***, whether it is inherent bias added when collecting or cleaning the data.

Hence, our first-pass proof of concept to validate the '*Don't Classify, Translate*' approach is to trust the integrity and the quality of the data and participate in the closed shared task scenario, where only the data provided can be used to train the model.

Now that we established a baseline model (`DCT mini`), compared it to an optimized version and a non-neural baseline and explored the obvious hyperparameter optimization options. We want to dig deeper into the dataset to understand how and when our model fail.

5.3 Uncertain Labels

Unlike a typical classification model where the last layer decides the most salient class label that the input should fall into, the DCT approach has an interesting by-product where it returns an empty string or a hallucinated string.

The following examples are some of the inputs on the FDI test set that `DCT mini` produced an empty label.

- *identifiez-vous*
- *Pour aller plus loin*
- *À lire aussi*
- *Un entretien*
- *Mais que l'on peut...*

There are a total of 744 empty labels produced by `DCT mini` on 22 unique text inputs in the test sets. It is worth noting *identifiez-vous* was repeated 714 times in the test set and *Pour aller plus loin* repeated 9 times.

These are 3 data points in the test set that produced hallucinating string as a label, the first *? ? ? ? ? ?* input appeared 8 times in the test data and the other are singleton occurrences.

- *? ? ? ? ? ?*
- *Quel est le seuil minimum d'acceptation pour que ça fonctionne ?* +
- +

To handle the above empty and hallucination situations, we simply fallback to the `FR` labels for these instances.

Technically, we could have looked at the n-best options produced in our beam search and look for the next best output that fits one of the label. However, leaving this bug/feature as is, we can use it to identify oddities in the training and validation data to improve the data quality.

5.4 Repetition in the Test Data

Academically, it makes sense to deduplicate the test set and report accuracy or F1-scores. Unless a test set is plagued with rampant repetitions, e.g. more than 30% of the test set are made up of repeats; from a user-experience perspective, ***deduplicating do no good to reflect the actual amount of errors a user experience*** when using the tool. It is best to leave the test data as if without deduplication if it is a random sample from the natural distribution of the full dataset.

Hypothetically, if the natural distribution of the input data has certain strings that repeats frequent, a user is more likely to report the error on the language label multiple times than sporadic errors that occurs once or twice. Thus, we view the repeated instances in the test set as a valid phenomenon and provide the following statistics solely to understand which instances are would cause the most user-dissatisfaction. Such scenario is evident in Table 3 where it shows 3 unique test instances repeating more than 100 times results in 4.2% of the test data.

No. of Times Repeated	No. of Unique Instances	% of Data
1	34,292	**93.35**
2	382	2.08
3	20	0.16
4 - 10	11	0.15
22	1	0.06
> 100	3	**4.20**

Table 3: Test Data Instances with Repeated Occurrences

Table 3 presents some statistics of repeated data in the test set. Of the 36,733 instances in the test set, 34,292 of them occurred once and 382 unique instances occurred twice. There are 3 instances that repeated $>$100 times, we have:

- *identifiez-vous* (714 times)
- *ici pour connaître la suite. déjà abonné ? identifiez-vous* (567 times)
- *déjà abonné ? identifiez-vous* (260 times)

Repeating the same exercise on training and development/validation dataset, Table 4 and 5 raises some alarm with 10-20% of the data repeating $>$50 times.

No. of Times Repeated	No. of Unique Train Instances	% of Data
1	234,518	**65.36**
2	40,745	22.71
3	1,547	1.29
4	4,97	0.23
5-50	75	0.30
> 50	172	**9.83**

Table 4: Training Data Instances with Repeated Occurrences

No. of Times Repeated	No. of Unique Dev Instances	% of Data
1	12,316	**68.41**
2	426	2.37
3	246	1.37
4	764	4.24
5-50	3298	**18.32**
>50	482	2.67

Table 5: Dev Data Instances with Repeated Occurrences

Given this knowledge of the repeated instances, the natural experiment to test is to deduplicate and/or remove the instances that >50 times and retrain the model to see if these data irregularities affected the weighted F1 performance of classification task. But that is out of scope of this report.

6 Related Work

While generic language identification seemed solved (McNamee, 2005; Lui et al., 2014; Xia et al., 2010), distinguishing language varieties which are often lower resourced remains a challenge (Fertmann et al., 2014; Tan et al., 2014; Zampieri et al., 2014, 2015). Hence, the language varieties identification task is a staple of the evaluation campaigns hosted by the VarDial workshops (Malmasi et al., 2016; Zampieri et al., 2017, 2018, 2019; Gaman et al., 2020; Chakravarthi et al., 2021). Across the many evaluation campaigns, probabilistic models like naive Bayes have often ranked top on the leaderboard (Bernier-Colborne et al., 2019; Bernier-Colborne and Goutte, 2020; Bernier-Colborne et al., 2021).

7 Conclusion

In this paper, we have described our experiments to reuse encoder-decoder transformer models as a classifier based on the "*Don't Classify, Translate*" idea. Evaluating on the French Dialect Identification (FDI) dataset, we found that a simple naive Bayes model works better than the 6 layers encoder-decoder models and a really small neural model worked even worse. And now, some concluding remarks:

> *The encoder-decoder transformer is a shiny hammer that works fairly well for many NLP/MT tasks. But note, the 'your miles may vary' (YMMV) caution. Also, as a sanity check, a simple non-neural approach is a good baseline.*

References

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. International Committee on Computational Linguistics (ICCL).

Gabriel Bernier-Colborne and Cyril Goutte. 2020. Challenges in neural language identification: Nrc at vardial 2020. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 273–282.

Gabriel Bernier-Colborne, Cyril Goutte, and Serge Léger. 2019. Improving cuneiform language identification with BERT. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 17–25, Ann Arbor, Michigan. Association for Computational Linguistics.

Gabriel Bernier-Colborne, Serge Leger, and Cyril Goutte. 2021. N-gram and neural models for uralic language identification: NRC at VarDial 2021. In *Proceedings of the Eighth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 128–134, Kiyv, Ukraine. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Gaman Mihaela, 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, Kiyv, Ukraine. Association for Computational Linguistics.

Tobias Domhan, Michael Denkowski, David Vilar, Xing Niu, Felix Hieber, and Kenneth Heafield. 2020. The sockeye 2 neural machine translation toolkit at AMTA 2020. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 110–115, Virtual. Association for Machine Translation in the Americas.

Susanne Fertmann, Guy Emerson, and Liling Tan. 2014. Language identification for low-resource languages. Technical Report for NLP projects for low-resource languages. Saarland, Germany.

Mihaela Gaman, Adrian Gabriel Chifu, William Domingues, and Radu Tudor Ionescu. 2022. FreCDo: A New Corpus for Large-Scale French Cross-Domain Dialect Identification. *(under review)*.

Mihaela Gaman, Dirk Hovy, Radu Tudor Ionescu, Heidi Jauhiainen, Tommi Jauhiainen, Krister Lindén, Nikola Ljubešić, Niko Partanen, Christoph Purschke, Yves Scherrer, and Marcos Zampieri. 2020. A report on the VarDial evaluation campaign 2020. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 1–14, Barcelona, Spain (Online). International Committee on Computational Linguistics (ICCL).

Yonatan Hadar and Erez Shmueli. 2021. Categorizing items with short and noisy descriptions using ensembled transferred embeddings. *arXiv preprint arXiv:2110.11431*.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Maggie Yundi Li, Stanley Kok, and Liling Tan. 2018. Don't classify, translate: Multi-level e-commerce product categorization via machine translation.

Marco Lui, Jey Han Lau, and Timothy Baldwin. 2014. Automatic detection and language identification of multilingual documents. *Transactions of the Association for Computational Linguistics*, 2:27–40.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 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. The COLING 2016 Organizing Committee.

Paul McNamee. 2005. Language identification: A solved problem suitable for undergraduate instruction. *J. Comput. Sci. Coll.*, 20(3):94–101.

Gillin Nat. 2016. Sensible at SemEval-2016 task 11: Neural nonsense mangled in ensemble mess. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 963–968, San Diego, California. Association for Computational Linguistics.

Andrew Ng. 2016. Nuts and bolts of applying deep learning.

Raymond Hendy Susanto, Ohnmar Htun, and Liling Tan. 2019. Sarah's participation in WAT 2019. In *Proceedings of the 6th Workshop on Asian Translation*, pages 152–158, Hong Kong, China. Association for Computational Linguistics.

Liling Tan, Marcos Zampieri, and Jörg Tiedemann. 2014. Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *In Proceedings of The 7th Workshop on Building and Using Comparable Corpora (BUCC*.

Khin Yee Mon Thant and Khin Thandar Nwet. 2020. Comparison of supervised machine learning models for categorizing e-commerce product titles in myanmar text. In *2020 International Conference on Advanced Information Technologies (ICAIT)*, pages 194–199. IEEE.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Fei Xia, Carrie Lewis, and William D. Lewis. 2010. The problems of language identification within hugely multilingual data sets. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. 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. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Ahmed Ali, Suwon Shon, James Glass, Yves Scherrer, Tanja Samardžić, Nikola Ljubešić, Jörg Tiedemann, Chris van der Lee, Stefan Grondelaers, Nelleke Oostdijk, Dirk Speelman, Antal van den Bosch, Ritesh Kumar, Bornini Lahiri, and Mayank Jain. 2018. Language identification and morphosyntactic tagging: The second VarDial evaluation campaign. In *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, pages 1–17, Santa Fe, New Mexico, USA. 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.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, and Jörg Tiedemann. 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. Association for Computational Linguistics and Dublin City University.

Marcos Zampieri, Liling Tan, Nikola Ljubešić, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the DSL shared task 2015. In *Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, pages 1–9, Hissar, Bulgaria. Association for Computational Linguistics.

Xuan Zhang and Kevin Duh. 2020. Reproducible and efficient benchmarks for hyperparameter optimization of neural machine translation systems. *Transactions of the Association for Computational Linguistics*, 8:393–408.