Simple models are all you need: Ensembling stylometric, part-of-speech, and information-theoretic models for the ALTA 2024 Shared Task

Joel Thomas, Gia Bao Hoang & Lewis Mitchell

School of Computer and Mathematical Sciences & Adelaide Data Science Centre,
The University of Adelaide, SA 5005, Australia

Correspondence: lewis.mitchell@adelaide.edu.au

Abstract

The ALTA 2024 shared task concerned automated detection of AI-generated text. Large language models (LLM) were used to generate hybrid documents, where individual sentences were authored by either humans or a state-of-the-art LLM. Rather than rely on similarly computationally expensive tools like transformer-based methods, we decided to approach this task using only an ensemble of lightweight "traditional" methods that could be trained on a standard desktop machine. Our approach used models based on word counts, stylometric features, readability metrics, part-of-speech tagging, and an information-theoretic entropy estimator to predict authorship. These models, combined with a simple weighting scheme, performed well on a held-out test set, achieving an accuracy of 0.855 and a kappa score of 0.695. Our results show that relatively simple, interpretable models can perform effectively at tasks like authorship prediction, even on short texts, which is important for democratisation of AI as well as future applications in edge computing.

1 Introduction

Detecting human- versus AI-generated content is important, for multiple reasons, including misinformation detection (Zhou et al., 2023), academic integrity (Kumar et al., 2024; Zeng et al., 2024), even healthcare records (McCoy et al., 2024). Increasingly, documents are likely to be hybrid-written, with portions of text being AI-generated, and potentially edited or augmented by humans. This introduces the challenge of authorship attribution of short texts such as individual sentences within a longer document, which can confound traditional approaches (Brocardo et al., 2013). The ALTA 2024 Shared Task is squarely focussed on this challenge, presenting a sentence-level authorship attribution task between human- and AI-generated sentences, where those sentences belong to a longer, hybrid-written document. Existing state-of-the-art approaches to this type of task are larger transformer-based, with models like SeqXGPT (Wang et al., 2023) and segmentation-based approaches (Lo et al., 2021) showing strong performance.

However, for many of the application domains above, there will likely be a desire to use "traditional" models for reasons of explainability and trustworthiness. Also, a current trend in machine learning is towards the use of lower-dimensional models, for reasons of speed, accessibility of data, explainability and ability to run "at the edge" such as on mobile devices. Motivated by this, and because we wanted to build on the existing large academic literature on authorship attribution, we opted to use "traditional" models such as those coming from stylometry, linguistics, and information theory. In order to experiment with a number of methods, we developed an ensemble approach comprising five such models. This ensemble model performed reasonably well on the held-out test set, with an accuracy of 0.855 and kappa score of 0.695. We hope our results demonstrate that relatively simple, interpretable models can perform well at distinguishing AI-generated from human-generated text, and that these models can still have relevance in a variety of application domains requiring explainable models.

2 Data

The full details of the shared task description can be found in (Molla et al., 2024). The task consisted of a training phase (phase-1) where models could be trained on a training set and tuned/tested on a development set via multiple submissions, and then a testing phase (phase-2) where a final model was assessed on an unseen held-out test dataset. Our training dataset comprised 212794 data points. Features included the ID (article ID), 'domain' (the

Proceedings of The 22nd Annual Workshop of the Australasian Language Technology Association, pages 207–212
December 2-4, 2024

domain the article belongs to, such as news, academic, etc.), the sentence to make predictions on, and the true label of the sentence.

The training dataset was class-imbalanced, with around two-thirds of its data points belonging to the 'machine' class and one-third to the 'human' class.

3 Methods

Our approach uses an ensemble of five separate models, the predictions of which are combined together to make an overall prediction.

3.1 Word counts model

This model uses TF - IDF (Term frequency - Inverse Document Frequency) to represent the sentences in the dataset. These vector representations are then classified into "Human" or "Machine" by a Naive Bayes Classifier. TF - IDF produces a sparse vector representing relative frequencies of tokens in a sentence. The Naive Bayes classifier uses this representation to classify sentences into "machine" / "human".

3.2 Stylometry model

This model uses a stylometric measure called "Burrows' Delta" to classify the sentences. Burrow's delta is used to compare stylistic distances between the texts (Evert et al., 2017). The starting point represents the text in a document as a bag of words. The word counts are then converted to relative frequencies to compensate for different text lengths. For further processing the n most frequent different words over the whole corpus is chosen. The word frequencies of all documents can be arranged as a document X words matrix at this stage after which word frequencies are standardised, ie, the word frequencies over the whole corpus is normalised such that their mean is 0 and standard deviation is one. This results in what is known as 'z-score', Z_i(D) = (f_i(D) - μ_i) / σ_i for word 'i' in document 'D'. The Burrows Delta Δ_B is calculated as a summation given by $\sum_{i=1}^{n} |z_i(D_1) - z_i(D_2)|$. For classifying a text as 'Machine' or 'Human', the burrows delta score for the two labels are compared. The label with a lesser delta (an indication of stylistic distance) is chosen as the predicted label for the text

3.3 Readability metrics model

Textstat[1] is a python library that helps extract statistics from text. It helps determine readability, complexity and grade level. We used 21 such metrics to represent each sentence in the dataset. This dataset with 21 readability metrics as features was dimensionally reduced using PCA techniques following which the dataset was reduced to 7 features that explained 96% of the variance in the data. This reduced dataset was trained on the K-nearest neighbours model with the 'k' value set to 5. Predictions were then made based on this model to classify each sentence as written by 'Human' or 'Machine'.

3.4 Part-of-speech model

Stanford CoreNLP (Manning et al., 2014), a natural language processing tool, is used to parse sentences and generate hierarchical part-of-speech (POS) structure trees. After parsing, we simplify each structure by retaining only the POS tags and discarding the hierarchy, focusing solely on the sequential tags representing each sentence's grammatical composition. These POS tags are then transformed into vector representations using term frequency (TF) alone, omitting inverse document frequency (IDF) due to the case-by-case nature of short texts where IDF is less impactful.

The resulting vectorized POS tag sequences are used as features to train a K-nearest neighbors (KNN) model, with the number of neighbors k set to 3. This KNN model is trained to classify sentences as being either 'Human' or 'Machine' generated, leveraging the POS tag patterns as distinguishing linguistic characteristics. Similar techniques to this have been deployed for related classification tasks, e.g., persuasion detection (Iyer et al., 2017).

3.5 Information-theoretic model

This model is based on the observation from previous works on authorship attribution that perplexity can be an effective indicator of authorship (Beresneva, 2016). We define a language model as the set of conditional probabilities $p(w|h)$, $h \in \mathcal{H}$, where h is the history of $n-1$ words before w, and $\mathcal{H}$ is the set of all sequences of length $n-1$ over a fixed vocabulary. The method then predicts the authorship of a particular text $T = \{w_1, w_2, ..., w_n\}$ given the histories h_a of a set of known authors a as the author having the lowest perplexity for $T|h_a$,

[1] https://pypi.org/project/textstat/

or equivalently, the lowest-entropy $H(T|h_a) = -\sum p(T|h_a) \log p(T|h_a)$.

Inspired by this, we use the following cross-parsed entropy rate estimator[2] $h(T||h_a)$ (Bagrow et al., 2019; South et al., 2022) to estimate the extent to which T can be predicted from histories h_a:

$$h(T||h_a) = \frac{n \log_2(n-1)}{\sum_{i=1}^{n-1} \Lambda_i(T|h_a)}, \quad (1)$$

where $\Lambda_i(T|h_a)$ is the longest subsequence starting at position i in the T that appears as a contiguous subsequence in h_a. This estimator has been studied in simulated contexts in (Bagrow and Mitchell, 2018; Pond et al., 2020) and tested on real datasets in (Smart et al., 2022). Here we use (1) at the character-level to predict authorship a from the author with the lowest $h(T||h_a)$.

3.6 Ensembling method

We explored two schemes for making a prediction based on the ensemble of input models: a simple weighting scheme and a random forest-based approach.

3.6.1 Weighted Vote

This simple ensembling method uses inputs from all the base models. The individual predictions of all the models were combined using a weighted vote, where each model is assigned a weight proportional to its 'kappa-score' when evaluated on the phase-1 test set.

3.6.2 Random Forest-based Stacking

Stacking is an ensembling method that combines the ability of different models to learn different parts of the problem to achieve a better-performing model than the individual models themselves. We used 4 models (all base models except the Part-of-Speech model) as part of this model.

Stacking involves 2 kinds of models, base models (Stylometric model, Word-counts model, Readability metrics model and Cross-entropy model in this case) and the meta-model (Random Forest in this case). The train data is split into two parts, training and validation sets. The base models train on the training set and make predictions for the validation set. Now at this stage, we have base model predictions as well as true labels for the data points in the validation set. The meta-model learns the relationship between the base model predictions and the true labels. Next, we will have the base models make predictions on the held-out test set and the meta-model will use those predictions and the relationship it had learned previously to arrive at predictions for the held-out test set.

[2] https://pypi.org/project/ProcessEntropy/

4 Results

Details of the shared task and the competition structure are in (Molla et al., 2024). Table 1 shows the base models' performance on the phase-1 test set in terms of both accuracy and the kappa score that was used for the competition. The information-theoretic model was the best-performing model with an accuracy of 0.847 and kappa score of 0.670. The other models performed comparably, with accuracies in the range of 0.670-0.747, and corresponding kappa scores between 0.273-0.512.

Table 1: Base Model Performances on the phase-1 test set.

Model Name	Accuracy	Kappa
Stylometric Model	0.670	0.273
Part-of-speech Model	0.720	0.389
Word Counts Model	0.747	0.512
Readability Model	0.742	0.432
Information-theoretic Model	0.847	0.670

Table 2 shows the models and the kappa scores achieved on the phase-2 test set. The Weighted vote model has performed slightly better than the Stacked model using Random Forest. Note that in both cases there appears to be a slight benefit in ensembling all models together over just using the best-performing information-theoretic model, demonstrating the value of combining the strengths of multiple models.

Table 2: Meta Model Performances on the phase-2 test set.

Model Name	Accuracy	Kappa
Weighted Vote	0.855	0.695
Stacking (RF)	0.853	0.684

The readability based model which initially had 22 features was reduced to 7 features using PCA. This was done because KNN performs better in low-dimensional space. Figure 1 shows the plot of first two principal components. While it is clear from the figure that both classes show a lot of overlap, it is also noteworthy that human points have

a more expanded spread compared to the machine class.

A similar trend is observed in plots between other principal components as can be seen in Figure 2 where we see that the machine data points seem to be concentrated in certain regions whereas the human data points expand out a bit more than the other class. This might suggest that the human style of writing can have more variability compared to that of AI.

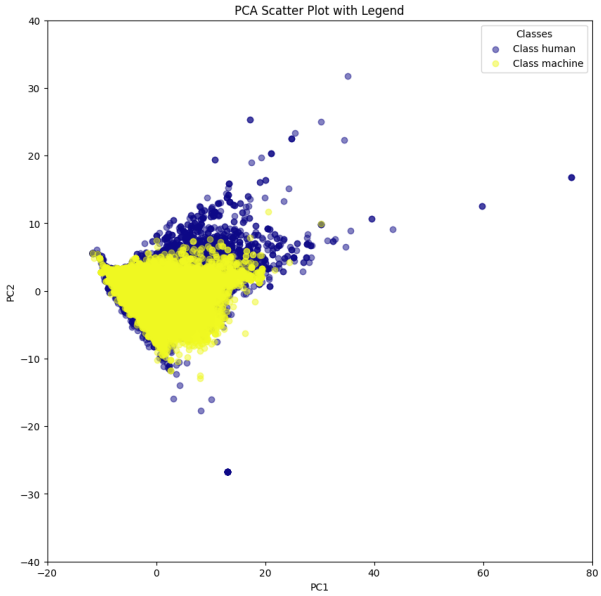

Figure 1: PCA of readability metrics.

5 Discussion

Our system was relatively simple, and therefore unlikely to ever achieve the highest scores in this Shared Task. Nonetheless, we think it performed very well, and demonstrates that simple models based on traditional methods can still be effective at distinguishing between human- and AI-generated text. How long this remains the case as generative large language models increase in sophistication remains an open question, however. Our approach had a number of limitations, which area left as future work. Firstly, we didn't consider the article structure, instead treating each individual sentence independently. This was partly in the interests of time, and because some methods used were less amenable to incorporating hierarchical structure than others. Hierarchical document structure could be incorporated in some methods, for example the naive Bayes model (Flach and Lachiche, 2004). We also did not always consider the domain of the document in the classification, for example in the information-theoretic model. This could be incorporated by splitting the documents in h_a based on domain, which might lead to an improvement in classification performance. Finally, we could consider each model's prediction confidence as part of the ensembling method. In the methods deployed here we only used the binary outcome predictions from each model as inputs to the ensembling method. However, incorporating a measure of the confidence of each model as inputs into the ensembling procedure is a more principled approach and has potential to improve the predictions, particularly in borderline cases where there might be disagreement between models. This would be straightforward to do for e.g., naive Bayes which produces probabilities as predictions, but would require the development of some heuristics for other methods, e.g., potentially using the difference in cross-parsed-entropy rates as a measure of prediction confidence for the information-theoretic model.

Acknowledgments

We wish to thank the organisers of the Shared Task for volunteering their time to create and run this Task. LM acknowledges funding from the Australian Research Council Discovery Project DP210103700.

References

James P Bagrow, Xipei Liu, and Lewis Mitchell. 2019. Information flow reveals prediction limits in online social activity. *Nature human behaviour*, 3(2):122–128.

James P Bagrow and Lewis Mitchell. 2018. The quoter model: A paradigmatic model of the social flow of written information. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(7).

Daria Beresneva. 2016. Computer-generated text detection using machine learning: A systematic review. In *Natural Language Processing and Information Systems: 21st International Conference on Applications of Natural Language to Information Systems, NLDB 2016, Salford, UK, June 22-24, 2016, Proceedings 21*, pages 421–426. Springer.

Marcelo Luiz Brocardo, Issa Traore, Sherif Saad, and Isaac Woungang. 2013. Authorship verification for short messages using stylometry. In *2013 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–6. IEEE.

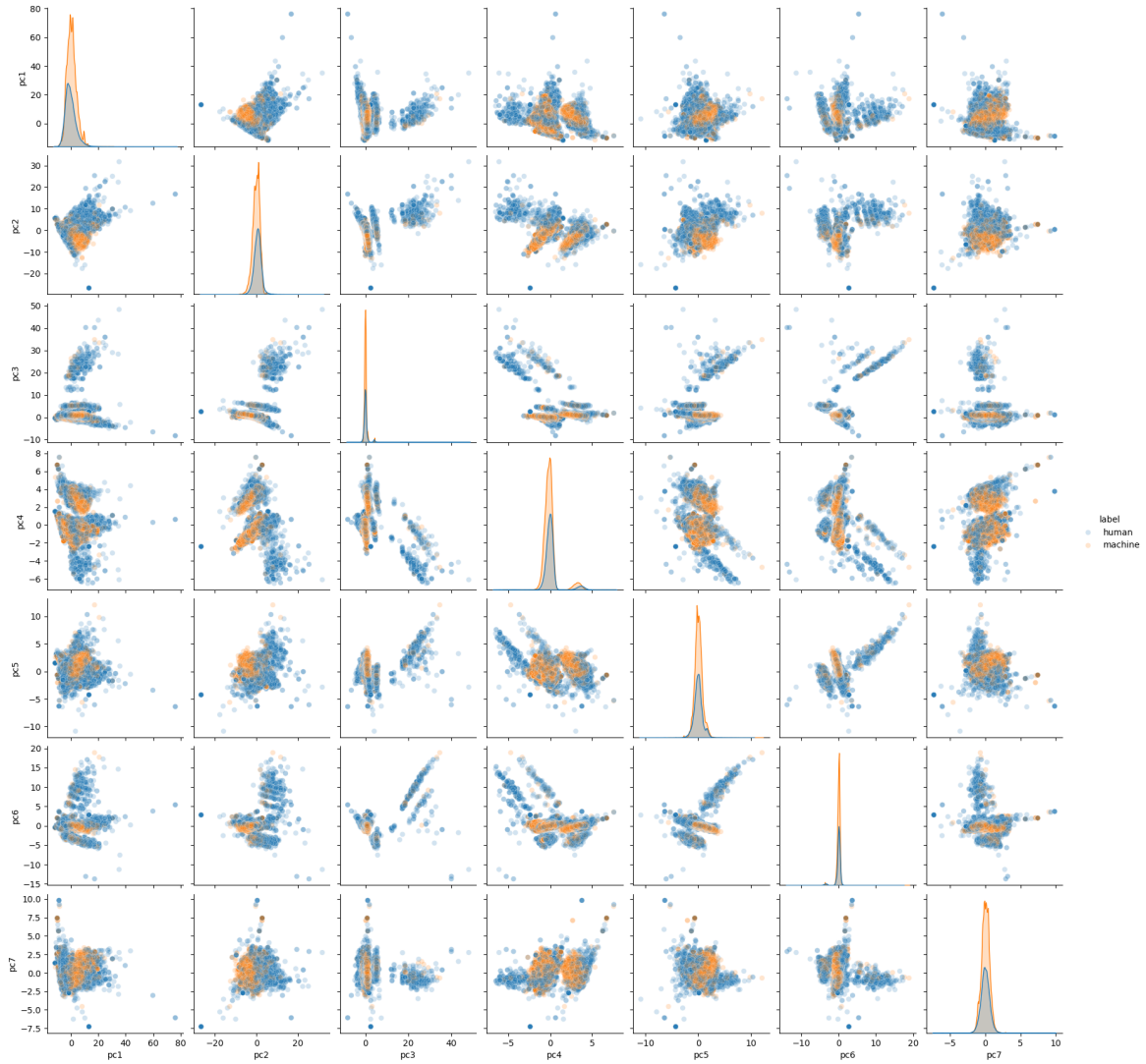

Figure 2: Pairplots using first 7 principal components of the readability models.

Stefan Evert, Thomas Proisl, Fotis Jannidis, Isabella Reger, Steffen Pielström, Christof Schöch, and Thorsten Vitt. 2017. Understanding and explaining delta measures for authorship attribution. *Digital Scholarship in the Humanities*, 32(suppl_2):ii4–ii16.

Peter A Flach and Nicolas Lachiche. 2004. Naive bayesian classification of structured data. *Machine learning*, 57:233–269.

Rahul R Iyer, Katia P Sycara, and Yuezhang Li. 2017. Detecting type of persuasion: Is there structure in persuasion tactics? In *CMNA@ ICAIL*, pages 54–64.

Rahul Kumar, Sarah Elaine Eaton, Michael Mindzak, and Ryan Morrison. 2024. Academic integrity and artificial intelligence: An overview. *Second Handbook of Academic Integrity*, pages 1583–1596.

Kelvin Lo, Yuan Jin, Weicong Tan, Ming Liu, Lan Du, and Wray Buntine. 2021. Transformer over pre-trained transformer for neural text segmentation with enhanced topic coherence. *arXiv preprint arXiv:2110.07160*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Liam G McCoy, Arjun K Manrai, and Adam Rodman. 2024. Large language models and the degradation of the medical record. *New England Journal of Medicine*.

Diago Molla, Qiongkai Xu, Zijie Zeng, and Zhuang Li. 2024. Overview of the 2024 ALTA shared task: Detect automatic AI-generated sentences for human-AI hybrid articles. In *Proceedings of ALTA 2024*.

Tyson Pond, Saranzaya Magsarjav, Tobin South, Lewis Mitchell, and James P Bagrow. 2020. Complex contagion features without social reinforcement in a model of social information flow. *Entropy*, 22(3):265.

Bridget Smart, Joshua Watt, Sara Benedetti, Lewis Mitchell, and Matthew Roughan. 2022. # istandwith-

putin versus# istandwithukraine: the interaction of bots and humans in discussion of the russia/ukraine war. In *International Conference on Social Informatics*, pages 34–53. Springer.

Tobin South, Bridget Smart, Matthew Roughan, and Lewis Mitchell. 2022. Information flow estimation: a study of news on twitter. *Online Social Networks and Media*, 31:100231.

Pengyu Wang, Linyang Li, Ke Ren, Botian Jiang, Dong Zhang, and Xipeng Qiu. 2023. Seqxgpt: Sentence-level ai-generated text detection. *arXiv preprint arXiv:2310.08903*.

Zijie Zeng, Lele Sha, Yuheng Li, Kaixun Yang, Dragan Gašević, and Guangliang Chen. 2024. Towards automatic boundary detection for human-ai collaborative hybrid essay in education. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22502–22510.

Jiawei Zhou, Yixuan Zhang, Qianni Luo, Andrea G Parker, and Munmun De Choudhury. 2023. Synthetic lies: Understanding ai-generated misinformation and evaluating algorithmic and human solutions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–20.