AutoAt: A deep autoencoder-based classification model for supervised authorship attribution

Anamaria Briciu, Gabriela Czibula, Mihaiela Lupea

Department of Computer Science, Babeș-Bolyai University Cluj-Napoca, Romania

Outline

Outline

Authorship attribution

Definition: Authorship attribution (AA) is the task of determining the likely author of a given text

Importance of domain: wide range of applications in:

- literature and history
- education
- social network analysis
- software engineering and cybersecurity

Proposed study

Exploit the ability of AEs to encode meaningful data patterns: we propose a **model based on an ensemble of deep autoencoders for authorship attribution**.

Dataset: poetic texts (language: Romanian)
Representation: document embeddings

Contributions

1. general classifier (proposed methodology easily applicable for texts from many domains)
2. distributed representation of poetic texts & model architecture (ensemble of AEs)
3. evaluation on a data set of poems authored by Romanian poets

Research questions

RQ1 How to introduce a multi-class classification model based on an ensemble of deep autoencoders to supervisedly identify the author of a given text, based on the encoded structural and conceptual relationships between the documents written by the same author?

RQ2 What is the performance of the approach introduced for answering RQ1 for identifying the authors of Romanian poetry and how does it compare to the performance of similar classification models?

RQ3 What is the relevance of the document embedding representation of the poetic texts in discriminating among different authors?

Outline

The authorship attribution task

- **Poetry**: [GC20] *(Language: Spanish; 5 poets; features: character n-grams)*, [AMM17] *(Language: Arabic; 73 poets; features: characters, word and sentence length, meter, rhyme, first word in sentence; algorithms: SVM, Naive Bayes)*, [GL19] *(Language: English; 5 poets; representation: bag-of-words; algorithm: SVM, Naive Bayes)*
- **Romanian texts**: [DPD08] *(2 Romanian novelists; features: frequency rankings of function words; algorithm: hierarchical clustering)*, [DN2] *(pastiche detection, extension of [DPD08])*

Methodology

- **using doc2vec** ([LM14]): [MHJ+17] *(social media texts; task: author profiling)*, [GAPDSP18] *(cross-topic authorship attribution)*
- **using autoencoders**: [STASH19] *(task: authorship verification; domain: cybercrime; texts: IRC messages; deep AE as one-class classifier)*, [MY07] *(AE-based one-class classification model for document retrieval task)*

Autoencoders (AE)

- deep learning models used in medical data analysis, image analysis, bioinformatics and other fields
- self-supervised learning technique

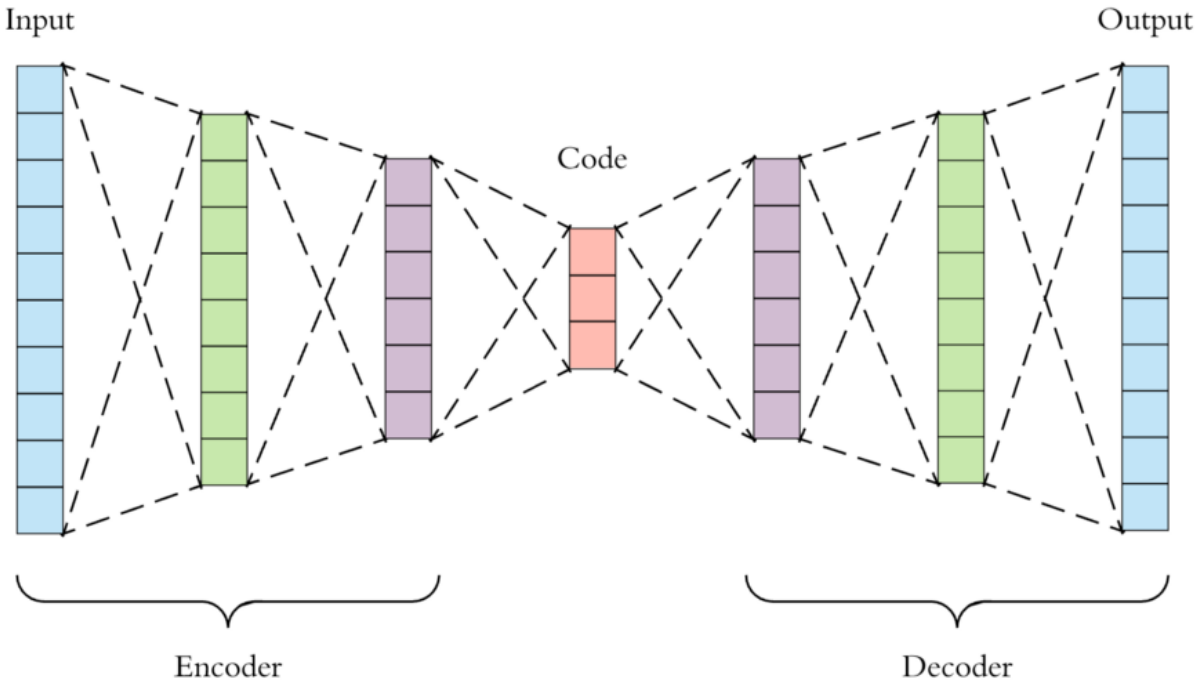

Figure: Autoencoder (AE) model[1]

[1] https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798

Outline

Formalization of the AA problem

Formalization as a **multi-class** classification problem.

- set of authors $\mathcal{A} = \{A_1, A_2, \ldots A_n\}$
- set of documents (texts) $\mathcal{T} = \{T_1, T_2, \ldots T_r\}$
- **GOAL**: approximate a target function $f : \mathcal{T} \rightarrow \mathcal{A}$ that maps documents from $\mathcal{T}$ to a certain class/author $a \in \mathcal{A}$.

AutoAt

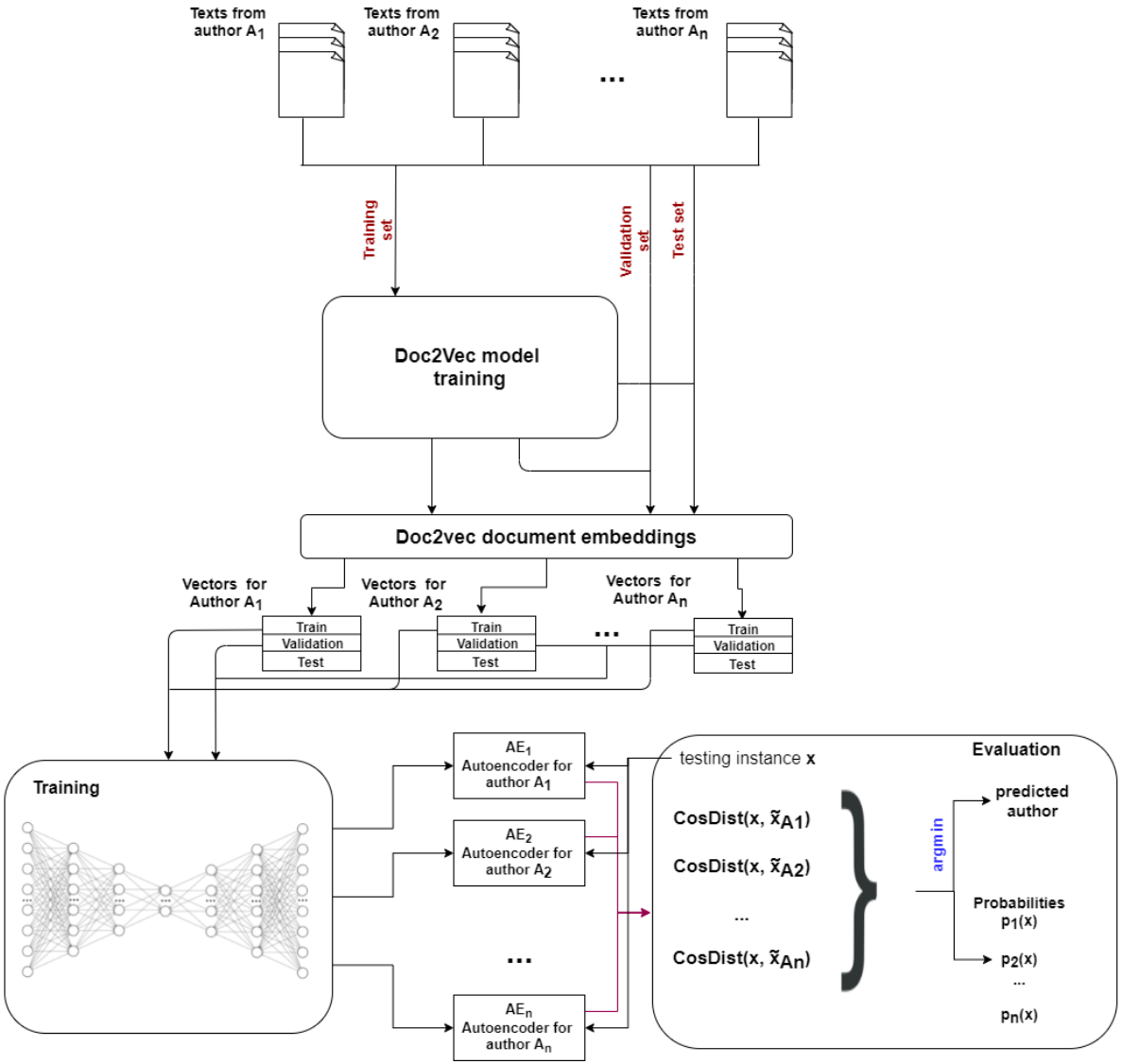

Figure: Overview of *AutoAt*.

The AutoAt model

- the *AutoAt* classifier consists of n autoencoders $AE_1, AE_2, \ldots, AE_n$, the autoencoder AE_i corresponding to the author A_i $(\forall 1 \leq i \leq n)$.
- AE_i will be self-supervisedly trained on the documents (texts) from $\mathcal{T}$ authored by the author A_i.

Data preprocessing & representation

Data preprocessing

Tokenization

Lemmatization of word tokens

Data representation

document embeddings obtained through the `doc2vec` model

Training (I)

A distinct autoencoder AE_i for each author A_i is trained.

Train-validation-test split

For each A_i $(\forall 1 \leq i \leq n)$, of D_i:

- 70% will be used for *training*
- 20% will be used for *validation*
- 10% will be used for *testing*

Loss function

$L(\tilde{x}, x) = \frac{1}{m} \sum_{j=1}^{m} (\tilde{x}_j - x_j)^2$

x represents the m-dimensional input

$\tilde{x}$ represents the model's m-dimensional output

Training (II)

AE architectures

- for `doc2vec` vectors of size 100: input_layer + 16-8-4-2-4-8-16
- for `doc2vec` vectors of size 150 and 300: input_layer + 128-32-16-8-4-2-4-8-16-32-128

Model details

- hidden layers use ReLU activation function
- encoding layer uses linear activation
- network trained using **stochastic gradient descent** + Adam optimizer
- mini-batch perspective (batch_size = 4)
- early stopping criterion - loss convergence on validation set is monitored (min_delta = 0.005)

Testing & evaluation: Classification (I)

For testing 10% from each data set D_i ($\forall 1 \leq i \leq n$) was used.

Classification

For test instance d:

- *AutoAt* searches for the autoencoder that minimizes the "distance" between d and $\tilde{d}$ (the instance reconstructed by the autoencoder).
- "distance" between 2 documents d_1 and d_2 defined as the **cosine distance** between them (where $cos(d_1, d_2)$ represents the *cosine similarity* between d_1 and d_2, scaled to [0,1])

$$CosDist(d_1, d_2) = 1 - cos(d_1, d_2)$$

;

Testing & evaluation: Classification (II)

- testing instance d
- $p_i(d)$ represents the probability that the input instance d belongs to class A_i
- $CosDist(d, \tilde{d}_{A_i})$ is the cosine distance between the instance d and its reconstruction $\tilde{d}_{A_i}$, through the autoencoder A_i

$$p_i(d) = \frac{1 - CosDist(d, \tilde{d}_{A_i})}{n - \sum_{j=1}^{n} CosDist(d, \tilde{d}_{A_j})}.$$

;

Testing & evaluation: Evaluation

$$Precision = \frac{\sum_{i=1}^{n}(w_i \cdot Prec_i)}{\sum_{i=1}^{n} w_i}$$

$$Recall = \frac{\sum_{i=1}^{n}(w_i \cdot Recall_i)}{\sum_{i=1}^{n} w_i}$$

$$F\text{-}score = \frac{\sum_{i=1}^{n}(w_i \cdot F - score_i)}{\sum_{i=1}^{n} w_i}$$

w_i = cardinality of D_i.

Outline

Dataset description

	Authors							
	Alexandru Macedonski	George Coșbuc	George Topîrceanu	Ion Minulescu	Mihai Eminescu	Octavian Goga	Vasile Alecsandri	Ștefan O. Iosif
ID	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
No. poems	190	212	113	159	366	181	186	164
No. tokens	39 403	124 809	31 525	35 380	182 270	37 761	72 025	30 870

Table: Description of data set

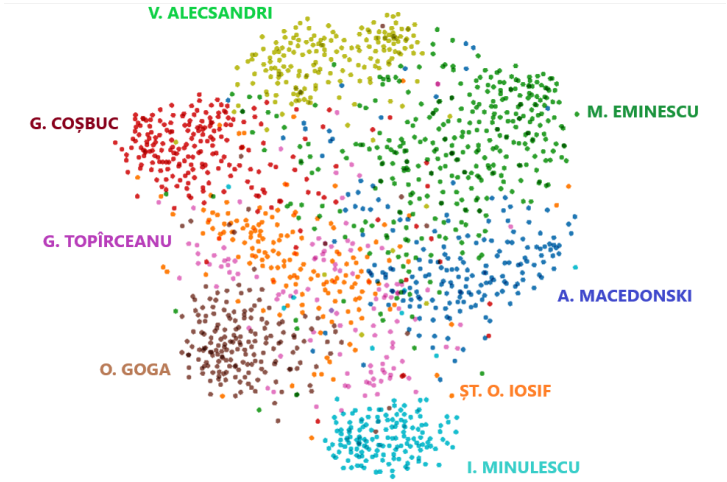

Figure: t-SNE [vdMH08] visualization of the data instances.

Results (I)

Number of features (m)	Performance measure	Authors								Overall
		A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	
100	*Prec*	0.85	0.83	0.58	0.89	0.92	0.73	0.84	0.67	**0.81** $\pm$ 0.017
	Recall	0.72	0.84	0.64	0.95	0.73	0.89	0.90	0.67	**0.79** $\pm$ 0.017
	F-score	0.77	0.83	0.67	0.92	0.82	0.81	0.86	0.66	**0.79** $\pm$ 0.018
150	*Prec*	0.88	0.85	0.63	0.89	0.93	0.73	0.82	0.68	**0.82** $\pm$ 0.019
	Recall	0.74	0.82	0.73	0.95	0.74	0.81	0.86	0.66	**0.8** $\pm$ 0.019
	F-score	0.80	0.83	0.68	0.92	0.82	0.81	0.86	0.66	**0.81** $\pm$ 0.02
300	*Prec*	0.82	0.83	0.62	0.91	0.98	0.68	0.79	0.73	**0.82** $\pm$ 0.015
	Recall	0.75	0.85	0.70	0.95	0.67	0.90	0.94	0.66	**0.8** $\pm$ 0.014
	F-score	0.78	0.84	0.66	0.93	0.79	0.77	0.85	0.68	**0.79** $\pm$ 0.014

Table: Experimental results. A 95% CI is used for the overall performance.

Results (II)

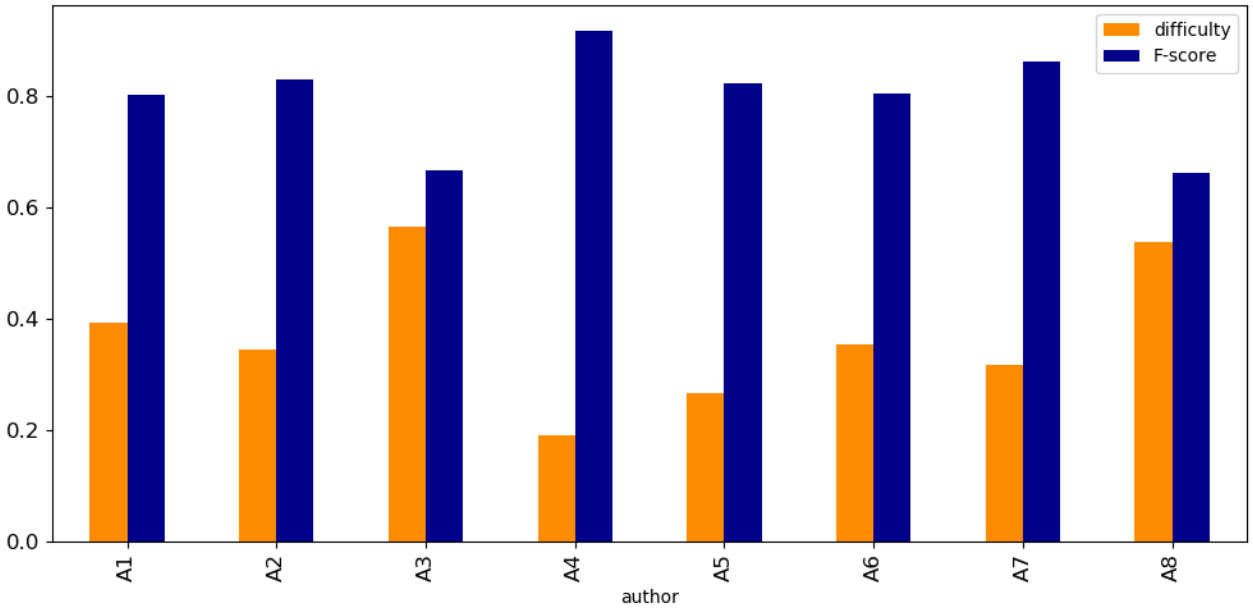

Figure: Correlation between the *F-score* values and the *difficulties* computed for each class/author.

Results (III)

Number of features (m)	Classifier						
	AutoAt	**SVC**	**MLP**	**LR**	**GNB**	**kNN**	**DT**
150	**0.81** $\pm$ 0.02	**0.83** $\pm$ 0.012	0.81 $\pm$ 0.017	0.78 $\pm$ 0.019	0.52 $\pm$ 0.027	0.41 $\pm$ 0.022	0.3 $\pm$ 0.024

Table: Comparison between *AutoAt* and classifiers from the literature in terms of *F-score*. 95% confidence intervals are used for the results.

Outline

Conclusions

- *AutoAt* classifier successfully solves the authorship attribution task for Romanian poetry
- document embeddings are appropriate representations that capture characteristics of authors (**future work**: combine doc2vec features with features specific to poetry)
- *AutoAt* is a general multi-class classifier, **future work**: investigate performance of *AutoAt* in other domains (e.g. source code AA)

Bibliography

A. Ahmed, R. Mohamed, and B Mostafa.
Machine learning for Authorship Attribution in Arabic poetry.
International Journal of Future Computer and Communication, 6(2):42–46, 2017.

L.P. Dinu, V. Niculae, and O. Șulea.
Pastiche detection based on stopword rankings. Exposing impersonators of a Romanian writer.
In *Proceedings of EACL 2012, Workshop on Computational Approaches to Deception Detection*, pages 72–77, 2012.

L.P. Dinu, M. Popescu, and A. Dinu.
Authorship Identification of Romanian texts with controversial paternity.
In *Proceedings of LREC 2008*, pages 3392–3397, 2008.

H. Gómez-Adorno, JP. Posadas-Durán, G. Sidorov, and D. Pinto.
Document embeddings learned on various types of n-grams for cross-topic Authorship Attribution.
Computing, 100:741—-756, 2018.

R. Guzman-Cabrera.
Author Attribution of Spanish poems using n-grams and the web as corpus.
Journal of Intelligent & Fuzzy Systems, 39(2):2391–2396, 2020.

C. Gallagher and Y. Li.
Text categorization for Authorship Attribution in English Poetry.
Intelligent Computing, 858:249–261, 2019.

Q. Le and T. Mikolov.
Distributed representations of sentences and documents.

In *Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 2014*, volume 32, pages 1188–1196, 2014.

I. Markov, Gómez-Adorno H., Posadas-Durán JP., Sidorov G., and Gelbukh A.
Author Profiling with Doc2vec neural network-based document embeddings.
Advances in Soft Computing. MICAI 2016, Lecture Notes in Computer Science, 10062:117–131, 2017.

L. Manevitz and M. Yousef.
One-class document classification via neural networks.
Neurocomputing, 70(7-9):1466–1481, 2007.

S. Shao, C. Tunc, A. Al-Shawi, and S. Hariri.
One-class Classification with Deep Autoencoder Neural Networks for Author Verification in Internet Relay Chat.
In *Proceedings of 16th IEEE/ACS International Conference on Computer Systems and Applications*, pages 1–8, 2019.

Laurens van der Maaten and Geoffrey Hinton.

Visualizing Data using t-SNE.
Journal of Machine Learning Research, 9(86):2579–2605, 2008.