Munibuc at Touché: Generalist Embeddings for Ideology and Populism Detection

Notebook for the Touché Lab at CLEF 2025

Marius Marogel[1,†], Silviu Gheorghe[1,†]

[1] University of Bucharest, Academiei 14, Bucharest, 010014, Romania

Abstract

Recent generalist text embedding models (gte) with customized instructions are designed to be used in many NLP applications such as classification, clustering, or retrieval. We use Nvidia's NV-Embed-v2 which has a Mistral-7b backbone with task-based instructions for Sub-Task 1 (ideology detection) and Sub-Task 3 (populism detection) to extract features for classification. Combined with a Support Vector Classifier, our system outperforms the proposed baseline and proves the reliability of tailoring generalist embedding models on various tasks on which they are not trained.

Keywords

generalist embeddings, customized instructions, LLMs for feature extraction, ideology detection, populism detection

1. Introduction

The political opinion of an individual can be seen as a point in a high-dimension vector space. People hold beliefs in regards to what are the most desirable public policies to be implemented, such as taxation and redistribution, in regards to ethical matters, spiritual opinions and so on. These possible choices can be seen as separate dimensions of the political space. These separate dimensions, however, are often quite well correlated, so, for example, the position of an individual in regard to taxation is quite a good predictor of the position related to other, seemingly unrelated matters, such as planned parenthood. Because of this fact, the political space can be mapped on lower dimension spaces such as a line or a plane. Examples of such spaces are the left-right spectrum [1] or left-right plus authoritarian-libertarian plane usually associated with the political compass. This represents a form of dimensionality reduction. The terms left and right have been coined and associated with the meanings we use today after the French Revolution (1789), based on the favourite physical spot that various parties took in the National Assembly. Another aspect, besides the politics to be implemented, is the strategy employed in taking or holding the power. Populism is sometimes seen as a strategy that aims to win votes by dividing the society into two opposing groups, the us vs them paradigm, usually the 'common people' vs 'the elites'. The political leaning and populism can be usually detected in a discourse due to the different use of the language associated with different politics and strategies.

1.1. Text classification

The proposed tasks can therefore be seen as text classification, an activity that's almost as old as the written language itself. We know that the library of Alexandria, for example, was organized into sections, according to the subject, in order to help the scholars find the work relevant to their field of

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

[†] These authors contributed equally.

marius.marogel@s.unibuc.ro (M. Marogel); silviu-florin.gheorghe@unibuc.ro (S. Gheorghe)
0009-0002-8957-8740 (M. Marogel); 0009-0002-0707-1218 (S. Gheorghe)

[1] There is also the opinion that this unidimensional space is better seen as being embedded in a two dimensional space, as in the horseshoe theory of politics, usually attributed to [1]

study. Text classification used to be a human tasks, but, with the advance of the digital representation of texts, it became central to the information technology. Early text classification techniques were based on boolean or statistic operations with the terms found in a document. They usually consisted of a 3-step process: Feature extraction that digitizes the text (usually in a high-dimensional vector space) followed optionally by dimensionality reduction and finally some kind of classification algorithm [2]. Most of the early methods were essentially bag-of-words methods, meaning that they didn't really consider the position of the words and the relationships between them. Using groups of N words (N-grams) was an early attempt to partially address this problem but the solution was only a temporary one. One of the first methods to change this was Long short-term memory (LSTM)[3]. These methods proposed a method to retain the context of the text seen so far, making sense of the words encountered in context, with practical applications in text translation sound processing and so on, arguably leading to the apparition of the modern attention mechanism some 20 years later.

A breakthrough in text processing happened with the invention of the modern quadratic transformers [4] and the attention mechanism. The method allows words to be finally understood in context. This permitted the apparition of pretrained language models (PLMs). PLMs are language models that are trained, usually in an unsupervised fashion, on a large corpus of text, allowing them to obtain a general understanding of the language and the world.

2. Related Work

The use of pre-trained models for text classification started when Yin et al. [5] proposed using the entailment problem[2] as a form of text classification. Specifically, to determine whether a text refers to a certain subject, say sports, the authors determine if the text to be classified, $\mathcal{T}$ entails a statement of the form *the previous text is about sports*. If the problem is multi-class such an inference is performed with each of them and the one with the highest confidence is presumed to be the correct class. Zhang et al. [6] determined that fine-tuning PLMs on various available tasks, where the task is described in a simple language, leads the models to the ability to perform simple language-described tasks. It was also shown that [7, 8] dividing the available NLP tasks into groups and fine-tuning them on some clusters improves the performance on other, unseen and unrelated clusters. Foundation models are PLMs that are pretrained on general data, with the specific purpose of being further adapted to various destination tasks. The general structure of the language is learned at the initial training and the specific details are learned later. The target task can be specified in two different ways: fine-tuning and intstruction tuning. Fine-tuning involves a training step from task-specific examples which can lead to catastrophic forgetting. Instruction tuning leverages the natural language task description during learning processes.

Political ideology identification is a reasonably well-studied task as shown by a recent survey [9]. Machine learning is often used to automatically classify the news. Especially of interest are the solutions submitted in the previous year as described in [10], due to the fact that they are address the exact same type of problem. In [11] multiple PLMs are used, forming an ensemble model. The work also makes use of data augmentation through back-translation. In [12] the authors use a fine-tuned BERT on the English translation of the parliamentary text. Other teams experimented with various classical Machine Learning models for feature extraction and classification such as TF-IDF, SVM, KNN and Deep Learning methods.

3. Methodology

The solution we propose for the Ideology and Power Identification in Parliamentary Debates 2025 shared task is an automated system consisting of two stages: feature extraction and classification. We customize a generalist embeddings model, Nvidia's NV-Embed-v2 [13], with task-based instructions for orientation and populism detection.

[2] for which some PLMs are pretrained

Generalist embedding models are a recent trend in Representation Learning with many applications in NLP. The *generalist* attribute of these models refers to their ability to capture relevant neural representations for a wide range of NLP tasks and subfields. Their performance is evaluated on massive benchmarks such as MTEB [14] (Massive Text Embedding Benchmark), which contain tasks such as text classification, clustering, retrieval, or sentence similarity.

We choose Nvidia's NV-Embed-v2 model to extract embeddings for both orientation and populism detection. NV-Embed-v2 is a generalist embedding model which ranked No. 1 on MTEB as of May 2024 and August 2024 and No. 5 as of May 2025. The authors of NV-Embed propose a lattent attention layer on top of pre-trained Mistral-7b [15], a decoder-only LLM, alongside a two-stage contrastive learning training process. They curate training data in the first stage for hard negatives and in-batch negatives and use contrastive learning only on retrieval datasets, while in the second stage they disable the use of in-batch negatives for the non-retrieval datasets.

We use custom instructions for the specialized embeddings on each Sub-Task on the English translation of the parliamentary texts and use them to create instructed queries as proposed by the authors in prompt 1:

- Orientation Instruction: "Classify the orientation of the political speech as either left or right"
- Populism Instruction:" Identify the position of the speaker's party on the populist–pluralist scale. Classify it as one of the following: 1 (Strongly Pluralist), 2 (Moderately Pluralist), 3 (Moderately Populist), or 4 (Strongly Populist)."

$$q_{instr} = \texttt{Instruct:\{task\_definition\} Query:} \{text\_en\} \tag{1}$$

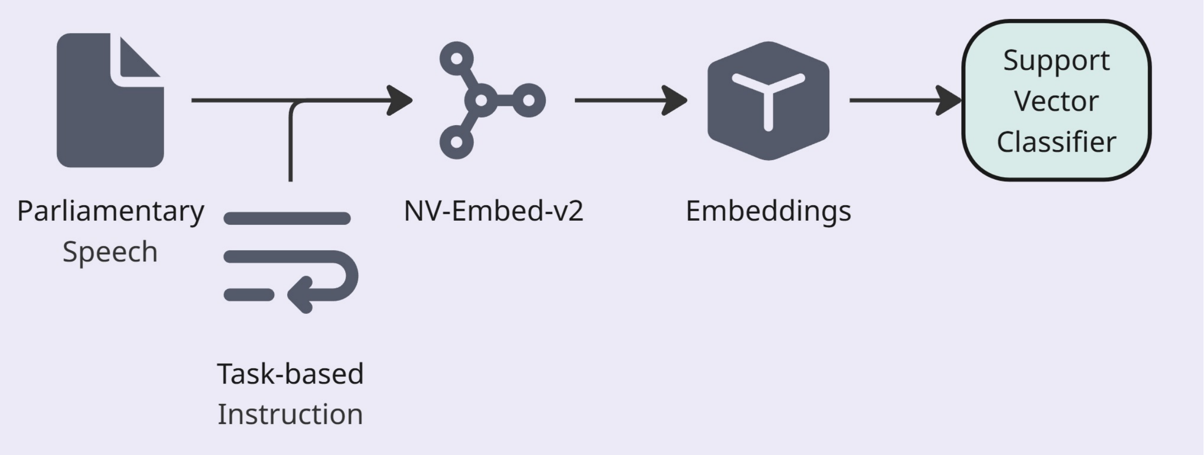

Figure 1: Proposed system with feature extraction from a generalist embedding model NV-Embed-v2 and a Support Vector Classifier

We use the resulted embeddings in a Support Vector Classifier (SVC) model for each task. We split the train embeddings into 5 splits to test the results of varying the C hyperparameter in the SVC implementation from scikit-learn [16] with *linear* kernel.

Hyper-parameter selection Parameter C, regularization, is varied between 10^{-2} and 10^{2} logarithmically, in 20 steps, separately for each parliament. The best value is computed using GridSearchCV, using five cross validation splits.

4. Results and Discussions

We evaluate the embeddings with SVC on a held-out validation set before training the final model for each parliament. Using a five-fold cross-validation and grid search, we determine the best hyperparameter

C in each country. We then compare the predictions on the held-out set with the predictions from the baseline model: a weighted tf-idf for feature extraction and a logistic regression (tf-idf+lr) for classification trained and evaluated on the same split.

Table 1
Sub-Task 1 - Validation (Orientation)

Model	AT	BA	BE	BG	CZ	DK	EE	ES	ES-CT	ES-GA	ES-PV	FI	FR	GB	GR	HR	HU	IS	IT	LV	NL	NO	PL	PT	RS	SE	SI	TR	UA
tf-idf+lr (baseline)	57	42	51	58	45	55	42	70	70	59	**100**	53	45	73	58	43	68	51	64	45	52	66	50	67	45	58	56	81	67
nv-embed+svc	**66**	**55**	**63**	**67**	**59**	**64**	**50**	**75**	70	**68**	49	**65**	**71**	**78**	**76**	**59**	**78**	**60**	**67**	**52**	**61**	**74**	**74**	**73**	**60**	**78**	**60**	**84**	**77**

Table 2
Sub-Task 3 - Validation (Populism)

Model	AT	BA	BE	BG	CZ	DK	EE	ES	ES-CT	FR	GB	GR	HR	HU	IS	IT	LV	NL	NO	PL	PT	RS	SI	TR
tf-idf+lr (baseline)	**57**	42	51	58	45	**55**	42	**70**	70	45	**73**	58	43	**68**	51	64	45	**52**	**66**	50	**67**	45	**56**	**81**
nv-embed+svc	53	**51**	**52**	**66**	**59**	50	**65**	68	**90**	**52**	42	**76**	**56**	63	**54**	**82**	**54**	43	55	**80**	51	**55**	55	77

Table 3
Sub-Task 1 - Test (Orientation)

Model	AT	BA	BE	BG	CZ	DK	EE	ES	ES-CT	ES-GA	ES-PV	FI	FR	GB	GR	HR	HU	IS	IT	LV	NL	NO	PL	PT	RS	SE	SI	TR	UA
tf-idf+lr (baseline)	60	44	60	57	41	57	39	69	66	74	45	57	46	77	61	46	58	57	66	42	51	67	50	66	47	59	61	79	49
nv-embed+svc	**66**	**54**	**64**	**67**	**56**	**65**	**53**	**73**	**67**	**84**	**46**	**62**	**69**	**83**	**70**	**57**	**73**	**65**	**69**	**57**	**63**	**72**	**75**	**73**	**56**	**82**	61	**80**	**55**

Table 4
Sub-Task 3 - Test (Populism)

Model	AT	BA	BE	BG	CZ	DK	EE	ES	ES-CT	FR	GB	GR	HR	HU	IS	IT	LV	NL	NO	PL	PT	RS	SI	TR
tf-idf+lr (baseline)	44	**37**	47	42	**58**	22	41	37	NA	33	50	57	33	29	21	NA	38	33	7	72	44	NA	**52**	65
nv-embed+svc	**49**	28	47	**53**	57	**44**	**51**	**57**	NA	**51**	**60**	**68**	**43**	**49**	**37**	NA	**40**	**37**	**20**	**79**	**54**	NA	51	**71**

Table 1 contains the comparison between our system (nv-embed+svc) and the baseline predictions on the held-out validation set for the orientation task (Sub-Task 1). We notice that our system outperforms the baseline tf-idf+lr on all parliaments, with the exception of es-ct with equal performance and es-pv with a very big difference which is due to there being no left-wing samples in the held-out validation and the model achieves 0.98 on right-wing and 0.00 on left-wing, thus the 0.49 macro F1.

We present the results of the held-out validation set in the populism task in Table 2. As we can see, for this Sub-Task there are multiple parliaments for which our method does not beat the baseline, with parliaments with a very high difference in performance. For example, in the pl parliament the difference is 30% and in es-ga the difference is 20% in our favor, while in gb the difference is 29% in favor of the baseline.

In Table 3, we present the results of the Orientation Sub-Task on the test set. NV-Embed-v2 features with SVC obtain an average of 0.66 macro F1, thus outperforming the logistic regression baseline of 0.57 macro F1. The same difference in performance is seen on the Populism Sub-Task in Table 4, where our system reaches 0.496 macro F1 compared to 0.418 macro F1 from the baseline. With the increase in performance of 8-9 points for both Sub-Tasks, we see the difference of tackling NLP tasks with customizable representations based on generalist text embedding models.

Future research. Given the results of NV-Embed-v2 with SVC on the test set, we consider generalist text embedding models as relevant feature extractors for both Sub-Tasks. Future experiments with different top-ranking models from MTEB are needed to assess the proposed methodology over a range of LLM-based embedding models. While we use pre-trained embedding models, fine-tuning them directly on the task or using contrastive learning techniques to retrieve political documents should uncover the full potential of generalist LLM-based text representations.

5. Conclusion

In this paper, we present the submission of team Munibuc for Sub-Tasks 1 and 3 (orientation and populism) for the Ideology and Power Identification in Parliamentary Debates 2025 shared task. Our approach is to use generalist text embedding models as feature extractors, thus evaluating the generalization capabilities of LLM-based embeddings on specific datasets. We extract task-based embeddings with customized instructions from a model based on Mistral-7b: NV-Embed-v2. Then, we use the extracted embeddings with a Support Vector Classifier with tuned hyperparameters on each parliament, resulting in an automated detection system which outperforms the baseline by a considerable margin on Orientation and Populism Sub-Tasks.

6. Acknowledgments

This work was supported by a grant of the Ministry of Research, Innovation and Digitization, CCCDI - UEFISCDI, project number PN-IV-P6-6.3-SOL-2024-0090, within PNCDI IV.

7. Declaration on Generative AI

During the preparation of this work, the authors did not use any Generative AI tool and take full responsibility for the publication's content.

References

[1] J. P. Faye, Théorie du récit: introduction aux langages totalitaires: critique de la raison, l'économie narrative, (No Title) (1972).

[2] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, D. Brown, Text Classification Algorithms: A Survey 10 (2019) 150. URL: https://www.mdpi.com/2078-2489/10/4/150. doi:10.3390/info10040150.

[3] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.

[4] A. Vaswani, N. Shazeer, N. e. a. Parmar, Attention is All you Need, in: Advances in Neural Information Processing Systems, volume 30, 2017. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[5] W. Yin, J. Hay, D. Roth, Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3914–3923. doi:10.48550/arXiv.1909.00161.

[6] S. Zhang, L. Dong, X. Li, S. Zhang, X. Sun, S. Wang, J. Li, R. Hu, T. Zhang, F. Wu, G. Wang, Instruction tuning for large language models: A survey (2024-12-01). URL: http://arxiv.org/abs/2308.10792. doi:10.48550/arXiv.2308.10792.

[7] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le, Finetuned Language Models Are Zero-Shot Learners, in: International Conference on Learning Representations, 2022. URL: http://arxiv.org/abs/2109.01652. doi:10.48550/arXiv.2109.01652.

[8] K. Shen, Z. Ju, X. Tan, et al., NaturalSpeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers, in: ICLR, 2024.

[9] T. M. Doan, J. A. Gulla, A survey on political viewpoints identification, Online Social Networks and Media 30 (2022) 100208.

[10] J. Kiesel, Ç. Çöltekin, M. Heinrich, M. Fröbe, M. Alshomary, B. De Longueville, T. Erjavec, N. Handke, M. Kopp, N. Ljubešić, et al., Overview of touché 2024: Argumentation systems, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2024, pp. 308–332.

[11] O. Palmqvist, J. Jiremalm, P. Picazo-Sanchez, Policy parsing panthers at touché: ideology and power identification in parliamentary debates, in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024). CEUR Workshop Proceedings, CEUR-WS. org, 2024.

[12] D. Chandar, D. Seshan, A. Koushik, P. Mirunalini, Trojan horses at touché: Logistic regression for classification of political debates (2024).

[13] C. Lee, R. Roy, M. Xu, J. Raiman, M. Shoeybi, B. Catanzaro, W. Ping, Nv-embed: Improved techniques for training llms as generalist embedding models, 2025. URL: https://arxiv.org/abs/2405.17428. arXiv:2405.17428.

[14] N. Muennighoff, N. Tazi, L. Magne, N. Reimers, Mteb: Massive text embedding benchmark, 2023. URL: https://arxiv.org/abs/2210.07316. arXiv:2210.07316.

[15] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, W. E. Sayed, Mistral 7b, 2023. URL: https://arxiv.org/abs/2310.06825. arXiv:2310.06825.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.