Original Manuscript

Social Science Computer Review
2024, Vol. 0(0) 1–23
© The Author(s) 2024
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/08944393241259434
journals.sagepub.com/home/ssc
Sage

Leveraging Open Large Language Models for Multilingual Policy Topic Classification: The Babel Machine Approach

Miklós Sebők[1], Ákos Máté[1], Orsolya Ring[1], Viktor Kovács[1], and Richárd Lehoczki[1]

Abstract

The article presents an open-source and freely available natural language processing system for comparative policy studies. The CAP Babel Machine allows for the automated classification of input files based on the 21 major policy topics of the codebook of the Comparative Agendas Project (CAP). By using multilingual XLM-RoBERTa large language models, the pipeline can produce state-of-the-art level outputs for selected pairs of languages and domains (such as media or parliamentary speech). For 24 cases out of 41, the weighted macro F1 of our language-domain models surpassed 0.75 (and, for 6 language-domain pairs, 0.90). Besides macro F1, for most major topic categories, the distribution of micro F1 scores is also centered around 0.75. These results show that the CAP Babel machine is a viable alternative for human coding in terms of validity at less cost and higher reliability. The proposed research design also has significant possibilities for scaling in terms of leveraging new models, covering new languages, and adding new datasets for fine-tuning. Based on our tests on manifesto data, a different policy classification scheme, we argue that model-pipeline frameworks such as the Babel Machine can, over time, potentially replace double-blind human coding for a multitude of comparative classification problems.

Keywords

quantitative text analysis, deep learning, large language models, classification policy agendas, comparative agendas project

Introduction

In the age of artificial intelligence (AI), along with many fields of social life, social science research is expected to be disrupted by technological advances.[1] Natural language processing

[1]HUN-REN Centre for Social Sciences, Hungary

Corresponding Author:
Miklós Sebők, HUN-REN Centre for Social Sciences, 4 Tóth Kálmán utca, 1097 Budapest, Hungary.
Email: sebok.miklos@tk.hun-ren.hu

(NLP) is situated at an intersection between information technology and linguistics, focusing on qualitative, textual data instead of using numeric data as inputs. It aims to bring automated solutions to various tasks, including machine translation, text summarization, or natural language generation (see chatbots). One such task is classification: the assignment of units to predetermined classes.

In the first wave of machine-assisted classification from the 2010s, dictionary-based methods and traditional machine-learning algorithms started to appear in research designs (Barberá et al., 2021; Hillard et al., 2008; Laver & Garry, 2000; Wilkerson & Casas, 2017). By the 2020s, in the second wave of leveraging machine learning techniques for various classification tasks, researchers could supplant human coding with automated solutions by utilizing a new crop of NLP resources: large language models (LLMs—on which many popular chatbot services are built)[2] (Bommasani et al., 2021; Devlin et al., 2019; Kaplan et al., 2020).

Despite these developments, many traditional international research projects in the field of comparative politics, such as the Manifesto Research on Political Representation (MARPOR) or the Comparative Agendas Project (CAP), are aimed at solving such classification tasks (Baumgartner et al., 2019; Baumgartner & Jones, 1991; Mikhaylov et al., 2012; Volkens et al., 2009). The CAP design is an example of a topic coding enterprise built around a codebook of predetermined classes. The codebook covers major policy topics (21 of them, from education to defense), and each unit of observation (covering a wide range of domains from newspaper articles to public opinion poll questions, laws, and budgets) is assigned precisely one such code.

Both MARPOR and CAP had long relied on large-scale human coding for database development. Yet, recent studies (Loftis & Mortensen, 2020; Máté et al., 2023; Navarretta & Hansen, 2022; Sebők et al., 2022; Sebők & Kacsuk, 2021) have successfully replicated human-level coding proficiency with machine learning for a wide variety of domains and data. Yet the application of LLMs is still in its infancy in comparative social research in general and for CAP-coding in particular (for exemptions, see Frantzeskakis and Seeberg (2023); Máté et al. (2023); Rytting et al. (2023)).

In this article, we set two objectives corresponding to a two-fold contribution to this literature. Our first objective was to offer a viable alternative to double-blind human coding for the CAP classification task regarding validity, reliability, and cost. We present a deep learning-based solution to the CAP classification task with the help of multilingual LLMs. We detail how so-called transformer models can be fine-tuned (trained) on diverse corpora (in terms of language and data source domain) to provide a fast, reliable, and inexpensive tool for automatically coding a large number of texts. Our second objective was to provide a technical description of our proposed solution, which can serve as a template for additional multilingual projects that tackle automated classification problems. To address this objective, we describe an open-source and free-to-use solution built around these models called the CAP Babel Machine. The CAP Babel Machine is a natural language processing system covering nine languages that aims to simplify and speed up research projects for scholars with limited technical backgrounds.

Our results show that using state-of-the-art multilingual XLM-RoBERTa large language models, the pipeline can produce gold standard-level outputs for nine languages with a potential coverage of over 100 languages. The weighted macro F1 of our models, a widely used metric for evaluating such solutions, trained on various language-domain pairs were, in 24 cases out of 41, above 0.75 (and, for 6 language-domain pairs, even above 0.90). This confirms LLMs as viable candidates to eventually replace human coders for the CAP task across a spectrum of languages, domains, and text features.

In what follows, we first provide an overview of the extant literature. Next, we present our research design for solving the CAP classification task. This is followed by a summary of the data used for fine-tuning the large language models and the pipeline for the automated production of policy topic labels. The section on Results presents model outputs according to language and domain. The Discussion explores three critical issues related to the usage of LLMs in comparative research: avenues for improving model performance, external validity and cost. The Conclusion weighs the comparative advantages (and drawbacks) of LLMs vis-á-vis human coding.

State-Of-The-Art Performance on the CAP Task

The validity of automated labels is traditionally measured against what is considered to be the gold standard (or "most reasonable benchmark" (Rauh, 2018, p. 7) in the literature for such research designs: the human coding of at least two well-trained coders (see double-blind human coding, with a high level of inter-coder reliability—e.g., White et al. (2015)). Reliability metrics are often presented simply in terms of inter-coder agreement (or joint probability of agreement), measured in percentages (e.g., Breunig and Schnatterer (2019) and McLaughlin et al. (2010)).

Krippendorff (2004) proposes a more refined and generalized metric that considers the number of coders, the number of categories, and the relationship between coding outcomes and those expected by chance. In addition, Cohen's kappa is a commonly used inter-coder reliability measure (Hemphill et al., 2019). In the Comparative Agendas Project (CAP) literature, Gava et al. (2017), Jungblut et al. (2023), and Kuipers and Timmermans (2021) present such metrics.

The international CAP codebook covers 21 "major" policy topics ranging from macroeconomics to public lands, with over 200 sub-topics ("minor" topics) for each major topic (see Bevan (2019, pp. 24–25) — in this article, we only focus on major topics). While slight deviations are common from the international standard, most projects use policy topic categories that conform to the comparative codebook. Inter-coder agreement or alpha levels are the most commonly shared for projects based on human coding. Krippendorff suggested that an acceptable intercoder reliability level comes at an alpha score higher than 0.8 (Krippendorff, 2004, p. 241). Proposed intercoder-reliability scores are also in this range.

For instance, the Dutch project requires a minimum of 95% intercoder reliability on the topic level (Timmermans & Breeman, 2019, p. 130), while the UK team observed a consistent range of 85–90% intercoder reliability (Bevan & Jennings, 2019, p. 178), while the Florida project aimed for a 90% agreement (Fahey et al., 2019, p. 204). Overall, K-alpha scores reported in the CAP literature are listed in the 0.8–0.9 range (Jungblut et al., 2023, p. 174; Kuipers & Timmermans, 2021, p. 365).

Extant computer-based machine-coding solutions to the CAP multiclass classification task experimented with both rule-based and ML-based approaches (as well as mixtures of them — Burscher et al. (2015), Frantzeskakis and Seeberg (2023), and Karan et al. (2016)). "Computer-assisted" (Collingwood & Wilkerson, 2012; Hillard et al., 2008; Lucas et al., 2015), "automated" (Flaounas et al., 2013; Quinn et al., 2006; Young & Soroka, 2012), or "semi-automated" (Breeman et al., 2009; Jurka, 2012; Kleinnijenhuis et al., 2013) studies have produced results that speak to the relative usefulness of these methods. On several classification tasks unrelated to CAP, LLMs soundly outperformed traditional machine learning (ML) algorithms (Conneau et al., 2019; Devlin et al., 2019). Nevertheless, LLMs have only been applied to the CAP classification problem in a few examples (see below).

For projects based on machine coding, Sebők et al. (2022, p. 3626) provided a detailed comparison (see Table 1). Therefore, we only reviewed and updated their summary here. The extant empirical literature—using various machine learning classifiers—on the CAP tasks exhibits an accuracy ranging between 60.5 and 82.2 (Barberá et al., 2019; Purpura & Hillard, 2006), precision values from 39.0 up to 71.3 (Albaugh et al., 2014; Barberá et al., 2021; Dun et al., 2021), while F1-scores when reported are around 0.70 (Burscher et al., 2015).

Table 1. Fine-tuning Dataset Composition by Languages.

Language	Observations	Percentage of Total
English	1,147,783	43.2
Hungarian	813,852	30.6
Danish	248,163	9.3
Spanish	225,686	8.5
Dutch	79,975	3.0
German	78,850	3.0
French	28,496	1.1
Portuguese	19,850	0.7
Italian	14,645	0.6

As for the limited number of studies using LLMs on the CAP task, Frantzeskakis and Seeberg (2023) used a BERT model to classify laws on the African continent and reached an F1 score of 0.84. Other preliminary work explored the GPT-3 model's performance using prompting to apply the CAP coding scheme to New York Times front pages and U.S. Congressional hearing summaries. Despite extensive prompt experimentation, the GPT-3 model could not surpass 55.0 and 60.0 accuracy on the two datasets, respectively (Rytting et al., 2023). Large language models also work well with translated texts. Máté et al. (2023) demonstrated that machine translation could be used to leverage transfer learning using LLMs and achieved a macro F1 score of 0.75 on classifying Polish laws into CAP major topics with a BERT model that was fine-tuned on English-translated Hungarian laws and bills.

Based on this review of the literature, there is no convergence on a single cutoff point that we could accept as standardized: we can broadly consider a 0.70–0.85 F1-score range as the current state of the art, depending on text domains, languages, modeling paradigms, average input text length (which differs vastly between tweets and laws), and data sizes. Even in a best-case scenario, precision levels (a key metric of validity) usually peak at the 80–90% level due to several factors.

These variables include the complexity of the underlying policy content of the units coded (such as newspaper articles or laws) and that of the codebook and rules (single code per observation). Due to these various factors, designating a single, all-purpose cutoff point would not even be desirable. Nevertheless, based on all available evidence, for the sake of simplicity, in our modeling work, we settled on a weighted F1 benchmark of 0.75 as the threshold for "research-grade" results.

Data and Methods

Although large-scale language models have revolutionized computer-based text analysis methods, the field is still dominated by studies based on English-language data. Pre-training LLMs (monolingual and multilingual models) and the limited availability of labeled data for a given task can prevent projects from achieving state-of-the-art results for low-resource languages. In sum, regarding the three critical aspects of comparative classification projects (validity, reliability, and various forms of cost), the lack of state-of-the-art classification *solutions* (as opposed to algorithms or trained models) is still a limitation for practical research.

We addressed these limitations in a multi-step process. First, we created a deep learning-based solution to the CAP classification task with the help of multilingual large language models. Second, we set up an open-source and free-to-use natural language processing system (called the CAP Babel Machine) built around these models trained for a concrete classification task.

This section presents the data used for fine-tuning and testing the model, model selection and training, and the pipeline in which the model generates predictions based on user uploads.

We procured data from two sources: publicly available datasets from the Comparative Agendas Project's website (comparativeagendas.net) and data kindly provided by the members of the CAP international community.[3] To our knowledge, all datasets were labeled by human coders and conformed with the strict quality requirements of the CAP project. These corpora constituted the training sets for fine-tuning LLMs and served as the gold standard for evaluating their performance.

In total, the training data contains 22 categories (21 CAP major policy topics and a "no policy content" category), nine languages (see Table 1), and ten domains (media, social media, parliamentary speech, legislative, executive speech, executive order, party manifesto, judiciary, budget, and public opinion) for a total of 2.66 million rows (see Table 1, and Table A1 in Appendix A).[4] Due to limited domain-level data, we opted to combine jurisdictions (such as the EU and the UK) into bigger pools based on shared languages (see Figure 1). As Table 1 shows, the language composition of the compiled database is imbalanced: it covers multiple languages, but English and Hungarian contents are overrepresented.

In the harmonized database (see Figure 2), we used 21 major policy topics and an additional "None" category for data without policy content (notably, not all data sources had instances for all policy topics). The imbalanced nature of the policy topic distribution of our final database is also

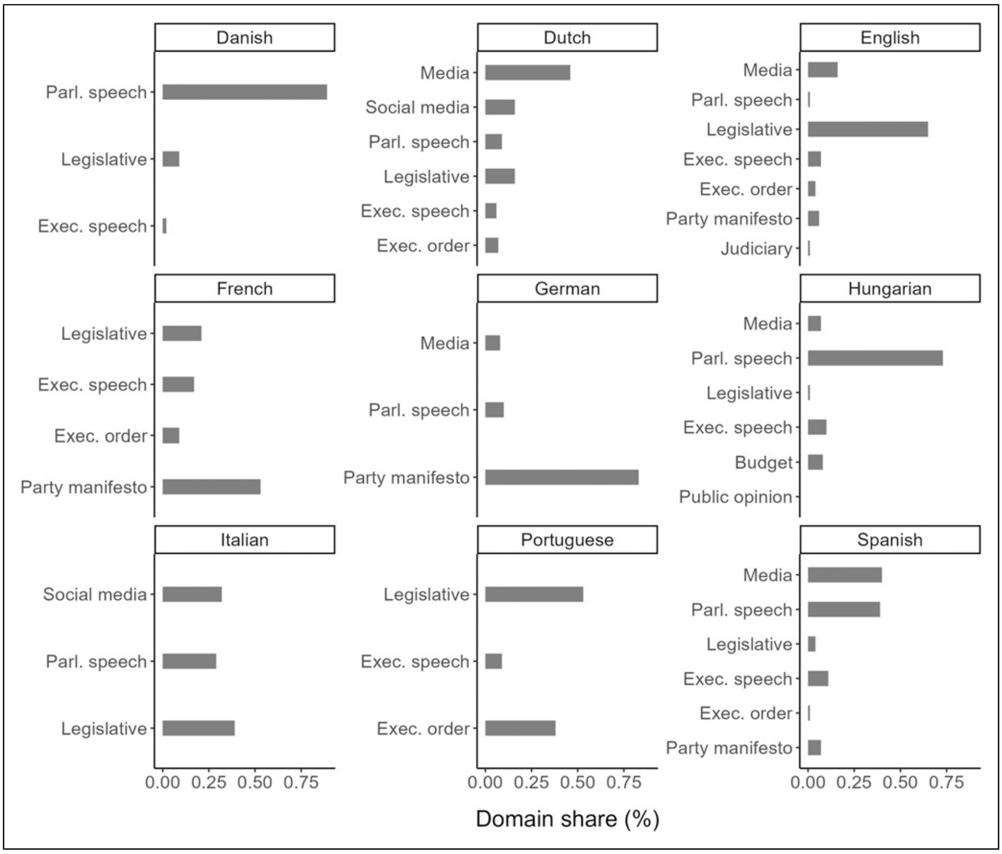

Figure 1. Distribution of domains by languages.

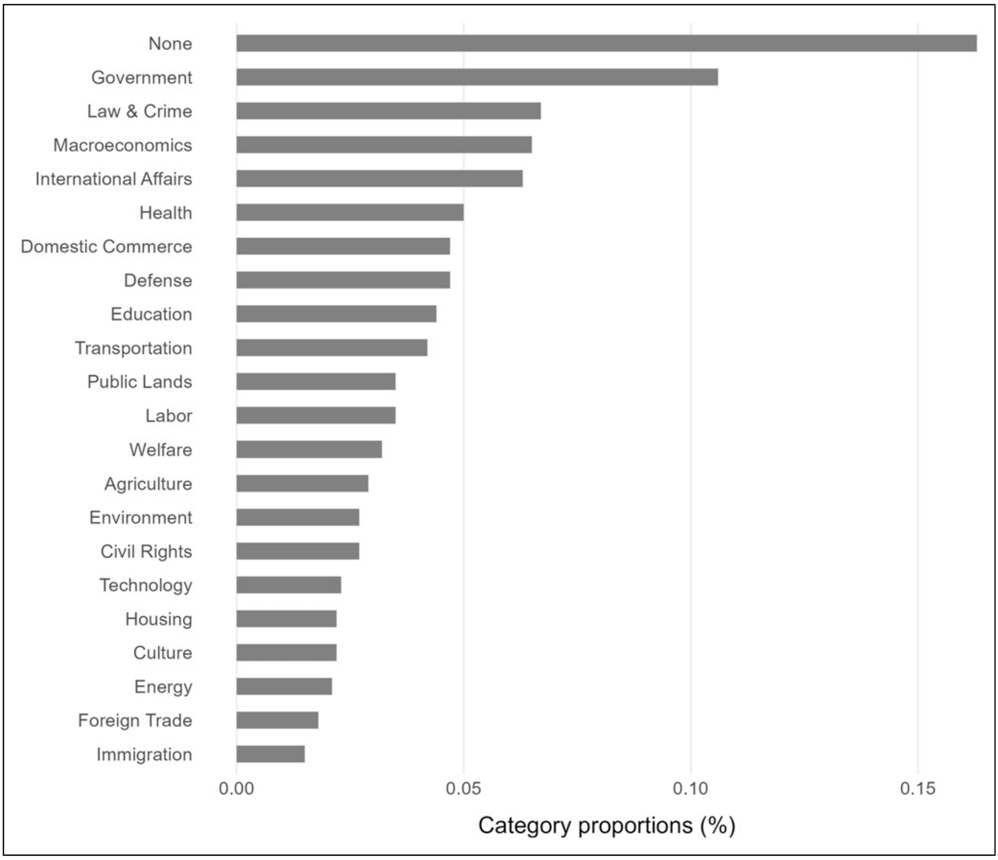

Figure 2. Category distribution in the pooled fine-tuning data.

well documented in the comparative agendas literature (Loftis & Mortensen, 2020; Sebők & Kacsuk, 2021).

We relied on fine-tuned large language models to predict CAP major topics on out-of-sample data. Current cutting-edge solutions use the Transformer architecture relying on the seminal work of Vaswani et al. (2017). The advantage of the Transformer architecture is that it can be combined with unsupervised learning for training (removing task-specific training) and the ability to fine-tune the models with domain-specific training data. Yet despite the proliferation of GPT-4 style commercial services, the open-source, multilingual large language model space remains limited.

Based on the literature and initial testing, we used the XLM-RoBERTa (large) model over smaller multilingual BERT models as it offers significantly better performance (Conneau et al., 2019).[5] The multilingual BERT model supports 104 languages (using 110 million parameters for the base model), whereas the XLM-RoBERTa model supports 100 languages (270 million parameters for the base model). For the unsupervised pre-training of XLM-RoBERTa, Conneau et al. (2019) used a filtered CommonCrawl database with over 2TB of data. This represents a significant increase in training data for low-resource languages over the multilingual BERT model.

As part of the pre-processing, we removed line breaks and superfluous whitespaces to clean the data (but otherwise left the texts as they were). Where applicable, we truncated the texts to 512 tokens and dropped those below five tokens to help with computational speed.[6] The resulting mean word count per observation was 63.2, with a standard deviation of 102 and a median of 23.

The XLM-RoBERTa model is similar to most transformer-based models in that it can only take 512 tokens as input. To access the pre-trained model checkpoint, we used the Huggingface repository, and for the fine-tuning, we used the Transformers library.[7] The parameters for the fine-tuning process were a batch size of 8 and a learning rate of 5e-6. We also used a dropout rate in the final classification layer of 0.1 to avoid overfitting the models.

The data for the fine-tuning was split into a training (80%) and test set (20%) by stratified sampling, where each stratum was calculated as the share of each source file (see Appendix A) within our final database containing all available data. In practice, all source data files were represented in the training and the out-of-sample test set with the 80/20 split. This sampling strategy ensured that all source data were represented proportionally in the fine-tuning data, regardless of language or domain. All the fine-tuned models were trained on a single NVIDIA A100 GPU with 80 GB RAM. In total, we fine-tuned 61 XLM-RoBERTa models: 1 pooled model (containing all data, for benchmarking purposes), 10 domain models (for strictly media data, etc.), 9 language models, and 41 language-domain (Danish-Executive speech) models.

Our evaluation metric for model performance was the weighted mean F1 score (also called macro F1), which accounts for the imbalanced data used for the fine-tuning. It captures model performance in a single value instead of relying on measures that might be easily inflated (accuracy) or only provide part of the performance picture (precision or recall). Macro F1 is the average of the class-based F1 scores. We weighted this average by the number of observations per category in the out-of-sample test data (this metric is also sometimes referred to as support) to account for label imbalance.

Our second contribution, besides fine-tuning LLMs for CAP classification, was to open up this resource to all interested parties at no cost (in terms of computing power or learning curve). The automated process for assigning CAP codes to uploaded corpora is based on the workflow shown in Figure 3. Users can upload their corpus through the babel.poltextlab.com webpage. The data is then moved into cloud storage, which triggers one of several large language model builds. The appropriate model out of the 29 options (selected based on the model-weighted macro F1 scores— see Figure 4) then automatically predicts policy topics for the text observations of the uploaded corpus. The models can handle a variety of languages, domains, and units of text (see above) and can be scaled up by adding new training data.

The labeling process is deterministic as the model runs in an inference mode, which turns off the stochastic process (e.g., dropout during the fine-tuning). This means that for the same input text, the same label is generated regardless of how many times it is uploaded. In addition to the predicted labels, the model also provides the softmax score of the three most likely categories. This additional information can be used to assess the confidence one should have in the assigned label.[8]

Results

A critical feature of any automation project is whether the underlying trained model is capable of producing research-grade results. Above, we defined an F1 score of 0.75 as a competitive performance level in terms of validity (which is a key feature of research design competitiveness besides reliability and cost—see Conclusion). We tested the respective performance of a total of 61 XLM-RoBERTa multilingual models against this benchmark. We fine-tuned the models on different training sets covering a pooled dataset (baseline model), one per each language (9), one for each language-domain pair (41), and one for each domain (10).

First, we fine-tuned a benchmark model on the pooled data containing all languages and domains (see Figure 5). The language-by-language performance of the model shows considerable variation, starting from weighted macro F1 scores of 0.83 down to 0.30. Notably, the language

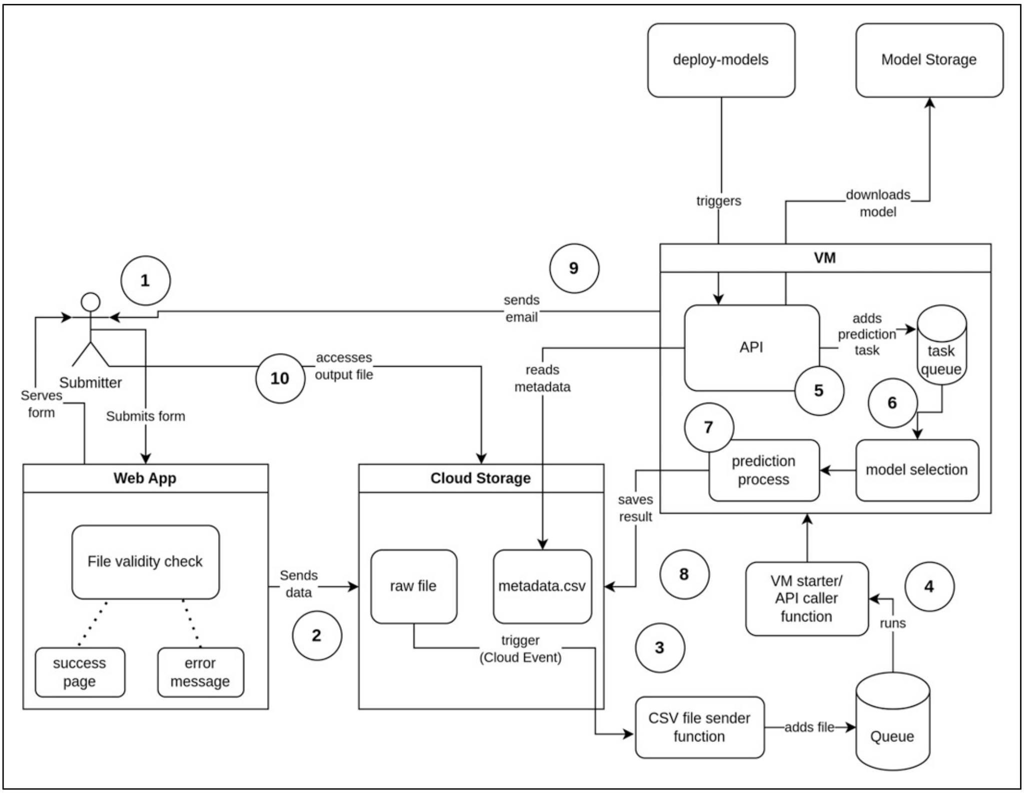

Figure 3. Flowchart of the CAP babel machine pipeline.

with the largest training set share (English) is in the middle of the pack with a weighted macro F1 of 0.59.

Some of the counterintuitive results may stem from the pooled fine-tuning dataset's multilingual and multi-domain (e.g., media, legal, and political) nature. Multilingual models must simultaneously learn the features and structure of multiple languages, and they might confuse language features. With single-language fine-tuning, the model focuses on the features of only one language and suffers from fewer confounding factors (Pires et al., 2019).

Therefore, in the second iteration, we trained language-specific models which are unique to each language in the database. Figure 5 shows our results, which show significant performance improvements (without exception) over the baseline model. Hungarian and Spanish data benefited the most from this approach: the Hungarian language macro F1 increased from 0.30 to 0.83, and the Spanish from 0.48 to 0.62. Overall, the results for some languages were already gold-standard quality, while others still underperformed vis-á-vis research-grade human results.

In a third step, we trained even more targeted XLM-RoBERTa models on language-domain subsets (e.g., English-legal, Dutch-media). The results in Table 2 show that this additional step also yields large improvements.[9] This is true with the caveat that variation between domains for the same language is still considerable. Excluding one Spanish outlier ("Parliamentary speech" with 0.38) and one Hungarian ("Budget" at 0.99), most results come in between 0.62 and 0.92 weighted macro F1.[10] On average, across languages, the "Legislative" domain has the highest model performance (with around 0.86 average macro weighted macro F1), followed by the "Executive order" (with 0.80).

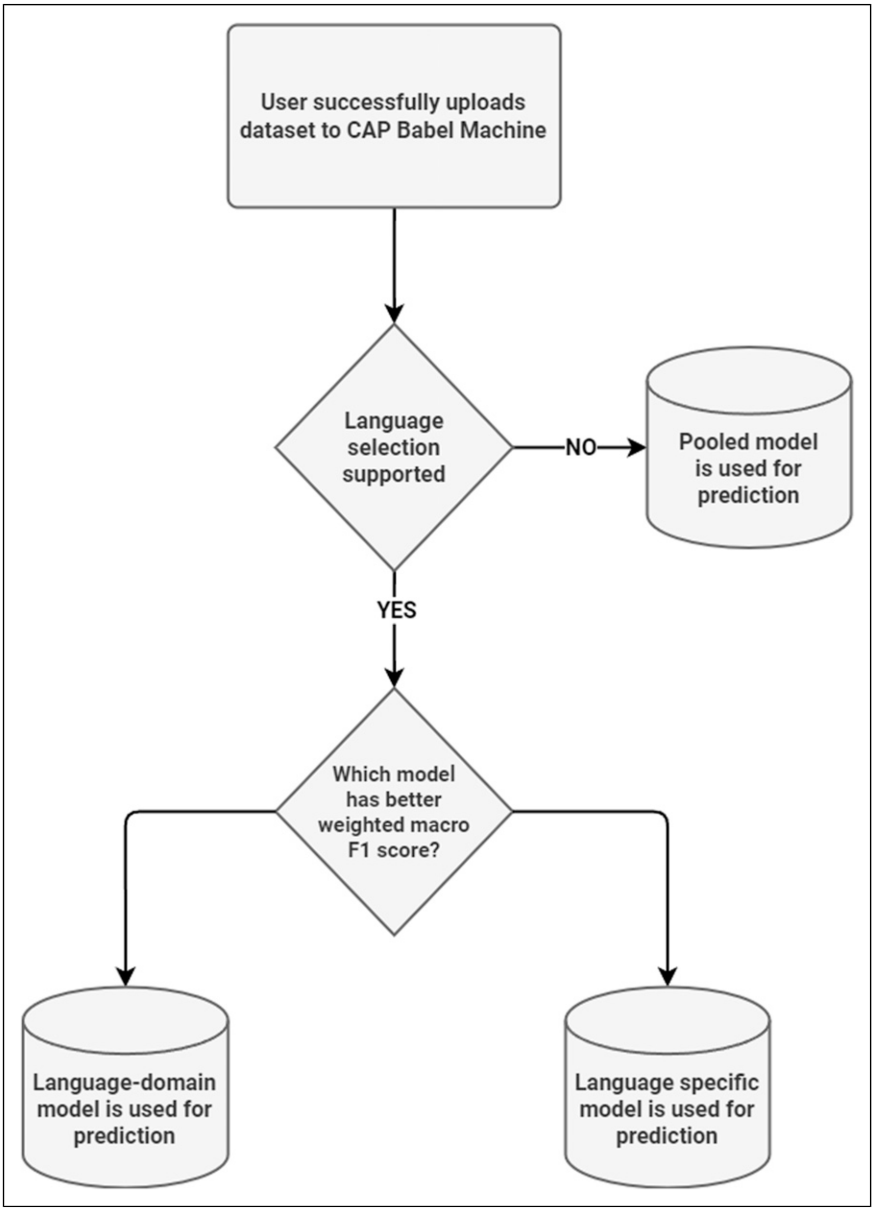
User successfully uploads dataset to CAP Babel Machine
Language selection supported
NO
Pooled model is used for prediction
YES
Which model has better weighted macro F1 score?
Language-domain model is used for prediction
Language specific model is used for prediction

Figure 4. Decision flowchart for prediction model selection.

Some domains are characterized by mediocre performance, such as the "Party manifesto" and "Executive speech" (at least compared to the other domain results). In the "Media" domain, a large variance is observable, with Dutch media achieving a 0.96 weighted macro F1 and German media with 0.62. Nevertheless, the results in Table 2 show state-of-the-art performance in many cases (a weighted macro F1 score above 0.75 in 24 out of 41 models and in 6 cases, even above 0.90).

Finally, exploring the language-domain models' performances on the category level shows that for most major topic categories, the distribution of micro F1 scores is centered around the 0.75 value (see Figure 6). The chart presents a micro F1 score distribution for 41 models

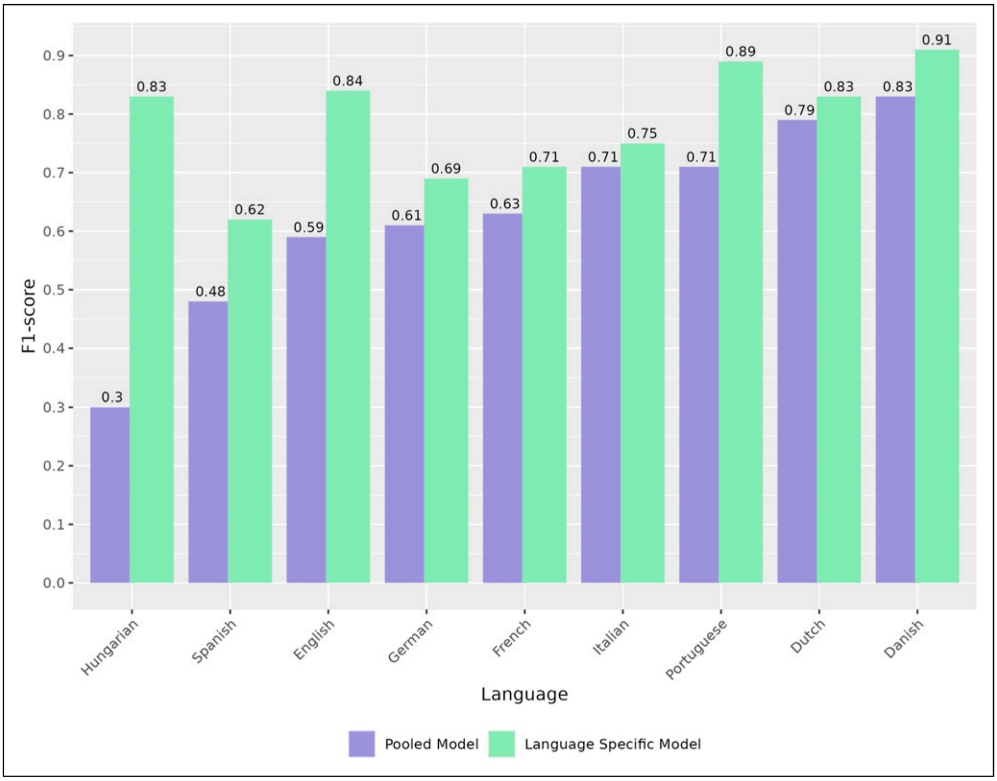

Figure 5. Performance of pooled versus language-specific models (weighted macro F1).

(language-domains) as smoothed density curves. The interpretation is similar to that of histograms: more frequent F1 scores have higher density scores and vice versa.

Some policy categories, such as “Health,” have a relatively normal distribution of F1 scores across models. However, in some cases, we observe long tails (e.g., “Public lands”) or multimodal distributions (e.g., “Foreign trade”). These might result from category overlaps between some policy categories or the uneven quality of training data. Nevertheless, Figure 6 shows that despite the often state-of-the-art results, there are variations in how models perform on a major topic level. The figure also features a vertical line to signal the average of the medians of the language-domain models. Finally, Figure 7 (based on Danish label-level data) shows no systematic patterns of mistaking one category for another.

Discussion

How to Improve Performance?

While, on average, our LLM-based results presented above are competitive with human coding and surpass those achieved with traditional machine learning, they still produce substandard results in many cases. One of the most straightforward solutions to improving model performance is leveraging more fine-tuning data for the given task. For the use case presented in this article, such a clear connection does not hold. This is evidenced by the discrepancy in the rank orders in Table 1 (training data size per language) and Table 2, which details model performance for data

Table 2. Performance by Language-Domain Pairs (Weighted Macro F1).

Domain											
Language	Pooled Domain	Media	Social Media	Parl. Speech	Legisl.	Exec. Speech	Exec. Order	Party Manifesto	Judiciary	Budget	Public Opinion
Danish	0.91	-	-	0.94 (0.89)	0.86 (0.09)	0.63 (0.02)	-	-	-	-	-
Dutch	0.83	0.96 (0.46)	0.8 (0.16)	0.79 (0.09)	0.84 (0.16)	0.66 (0.06)	0.77 (0.07)	-	-	-	-
English	0.84	0.78 (0.16)	-	0.82 (0.01)	0.9 (0.65)	0.71 (0.07)	0.78 (0.04)	0.73 (0.06)	0.76 (0.01)	-	-
French	0.71	-	-	-	0.85 (0.21)	0.80 (0.17)	0.73 (0.09)	0.66 (0.53)	-	-	-
German	0.69	0.62 (0.08)	-	0.72 (0.1)	-	-	-	0.71 (0.83)	-	-	-
Hungarian	0.83	0.69 (0.07)	-	0.84 (0.73)	0.85 (0.01)	0.65 (0.1)	-	-	-	0.99 (0.08)	0.93 (0.00)
Italian	0.73	-	0.62 (0.32)	0.65 (0.29)	0.81 (0.39)	-	-	-	-	-	-
Portuguese	0.89	-	-	-	0.93 (0.53)	0.71 (0.09)	0.88 (0.38)	-	-	-	-
Spanish	0.62	0.76 (0.40)	-	0.38 (0.39)	0.85 (0.04)	0.71 (0.11)	0.85 (0.01)	0.75 (0.07)	-	-	-

Note. The pooled-domain models in Column 2 are the same as the language-specific models in Figure 5. The domains' share within each language corpora is in parenthesis. Small shares (e.g., 0.001) are rounded to 0.0.

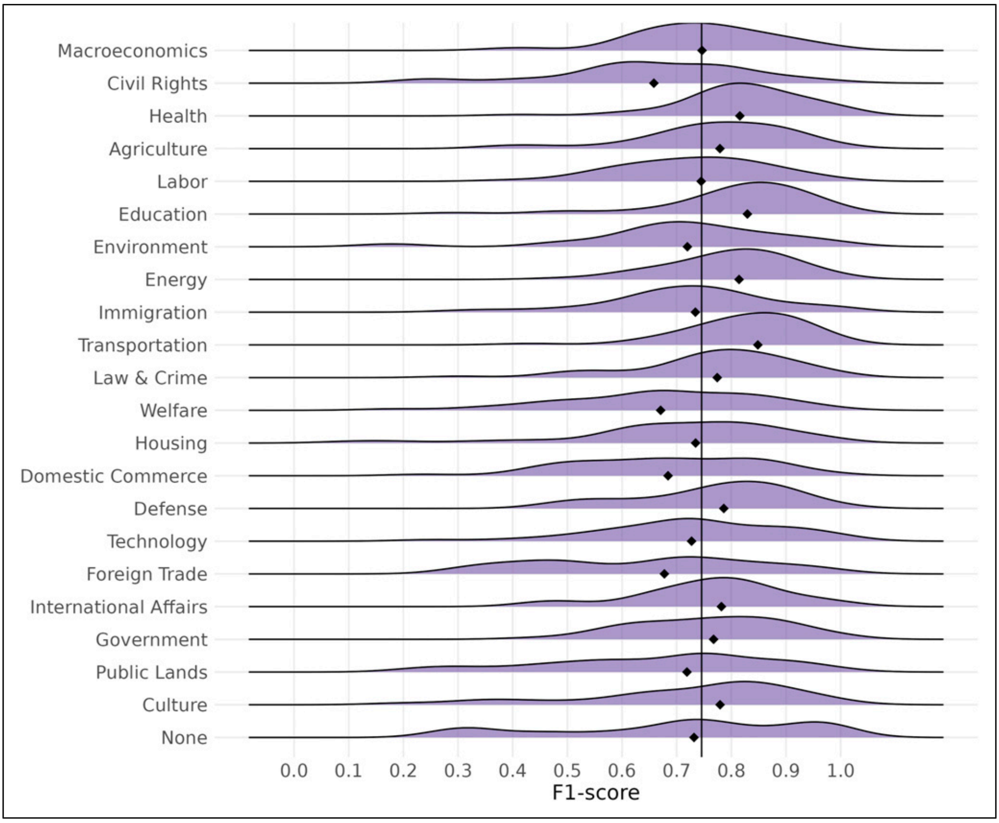

Figure 6. Distribution of micro F1 scores and their medians across CAP topic.

pooled across domains. English language models are in the middle of the pack despite having been fine-tuned on data an order of magnitude larger than other models for other languages, such as Dutch or French.

What seems more critical is to train separate language-domain models with suitably large fine-tuning data for each such model. Yet, training data size is not a get-out-of-jail card, even in these cases. As Table 2 above and Figure 8 below show, there are no readily discernible clusters of languages, sample sizes, or domains that would affect the fine-tuned model performance. Most language-domain pairs have a relatively low share (<30%) in the training data and a weighted macro F1 between 0.75 and 0.85. This warrants further analysis of additional factors.

One such area of improvement is related to the quality of input data used for fine-tuning. To tease out its potential impact on performance, we ran tests on our original Hungarian media corpus (containing 88,319 front-page articles from 1990 to 2014). After discussing data quality issues with the creators of this corpus (one of the laggard datasets when it comes to F1 scores), we removed one government cycle-worth of input data (2010–2014) where inter-coder reliability was lowest. As a result, model performance improved from 0.49 to a 0.69 weighted macro F1 score despite dropping 21,450 coded articles. Based on this evidence, data quality may be a critical explanatory factor of below-average model performances.

A third area of improvement is related to data homogeneity. Our approach (based on language-domain pairs) can be refined by introducing jurisdiction-language-domain models. This would affect current results for English (for which American, British, and EU data was used—see

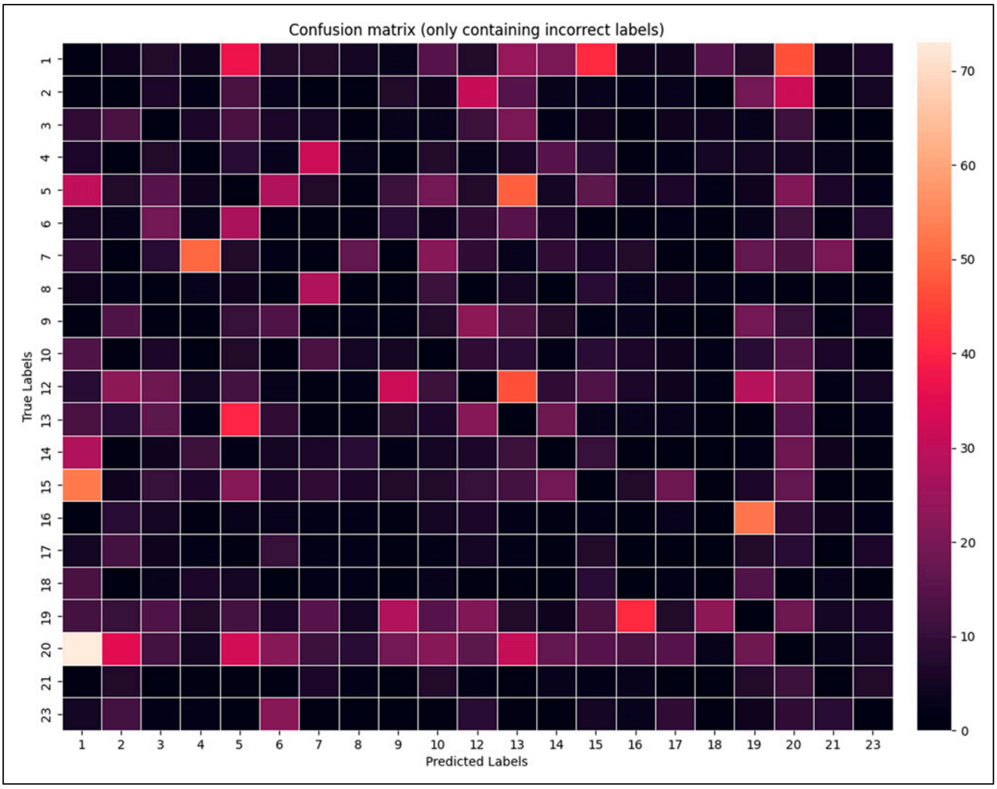

Figure 7. Confusion matrix for the incorrect labels (for Danish data).

Appendix A). Nevertheless, such disaggregation would further limit sparse fine-tuning data for various domains. Fourth, in contrast to traditional machine learning algorithms, data cleaning and feature selection is less of an issue as LLMs require no excessive pre-processing, as it reduces useful information in fine-tuning. Discarding extra whitespace and row breaks may have some effect, but overall, standard pre-processing procedures and handling class imbalance are not proven to have an impact (Máté et al., 2023, pp. 32–33).

A fifth layer of model performance quality checks can focus on class-level heatmaps generated from confusion matrices of language-specific models (see Figure 7 above). Our tests show no cross-lingual trends of misclassified codes, and given the low counts for mispredictions, they are unlikely to significantly impact overall, weighted F1 scores. Finally, one way to increase the predictive performance of models is to deploy additional techniques, including hyperparameter tuning (see our results related to Spanish speech data) and ensemble voting (where the results from multiple models are aggregated into a "majority" prediction).

Introducing elements of active learning (Miller et al., 2020) into the research design is also an option. In such a process, a new subset of machine-labeled data is selected for the worst-performing classes for in-process validation purposes (Sebők & Kacsuk, 2021, p. 36). After validation, the newly acquired hand-coded observations are integrated into the training corpus, augmenting the training dataset's quality and quantity (Sebők et al., 2022). However, it is important to note that this process incurs additional labor costs and requires more time and computational resources than standard learning processes.

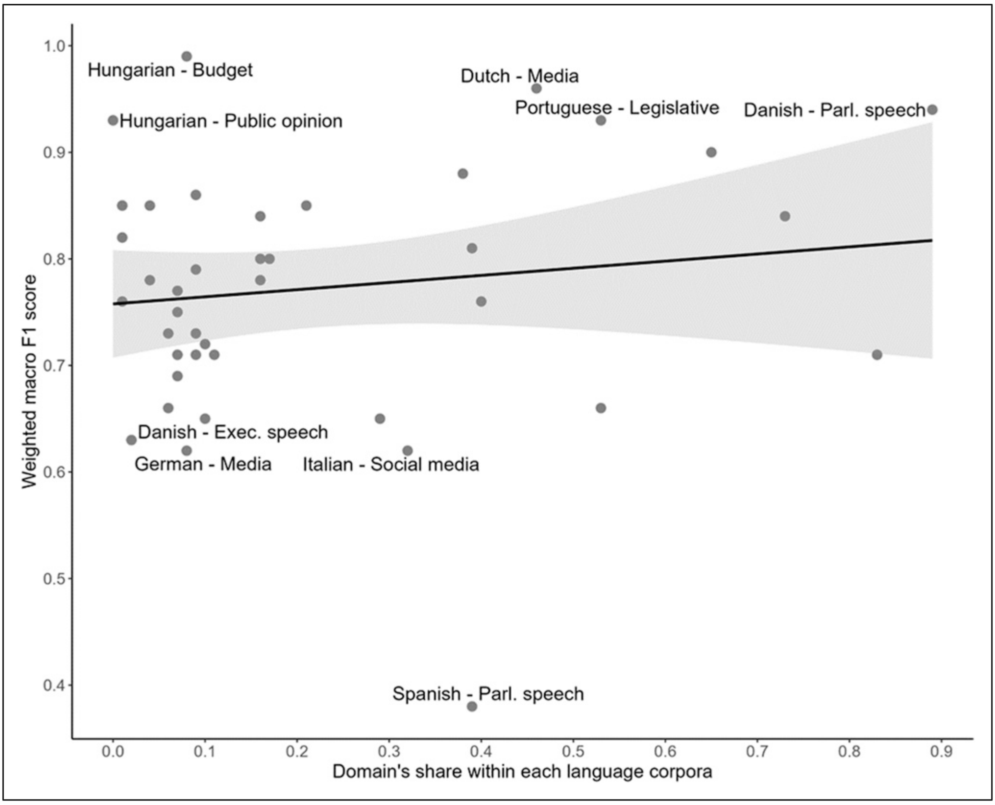

Figure 8. Domain share of language corpora and model performance. Note: Language-domain models with above 0.90 and below 0.65 weighted macro F1 are labeled. The shaded area represents the 95% confidence interval. The fitted OLS regression line's equation is y = 0.758 + 0.067x, with an R squared of 0.019, where y is the weighted macro F1 score, and x is the domain's share within each language corpora. The estimated effect of the domain's share is not significant.

External Validity

The CAP Babel Machine was designed with an eye toward generalizability. We chose the Comparative Manifesto Project/MARPOR to test its external validity. It uses a category system whereby each quasi-sentence of every manifesto is coded into one and only one of 56 standard categories. The 56 categories are grouped into seven major policy areas and are designed to be comparable between parties, countries, elections, and across time (Lehmann & Zobel, 2018; Merz et al., 2016).

To buttress our claim for generalizability, we applied the CAP fine-tuning and inference workflow to Manifesto data as a proof-of-concept exercise. We downloaded the English, Norwegian, and Spanish labeled data via the manifestoR package and created a training/test split of 80/20 (Lewandowski et al., 2020). In the next step, we fine-tuned an XLM-RoBERTa model using the same parameters as we did for the CAP models. Without additional modeling work (which was extensive for achieving state-of-the-art CAP results), our average Manifesto results were still above 0.60 (weighted macro F1) for the smaller languages and 0.55 for English data (see Table 3).

The inferior results compared to CAP models are also a function of the significantly longer category list of the Manifesto project (where the probability of randomly choosing the correct label is 1/56 + 1, where the extra label is 0). Furthermore, micro F1 performance varies significantly as the models achieved an F1 close to 0.80 for some categories (the minimum results were omitted as, for technical reasons, they had a value of 0 for multiple classes due to a low number of observations). As for the front end of the pipeline, adding an alternative coding scheme only requires a single additional roll-down menu for choosing the classification task and associated codebook. The model selection process would be simplified on the backend as manifestos constitute a single domain. While these results are only preliminary, they demonstrate the viability of our pipeline and its applicability to additional classification tasks, provided there is ample training data for model fine-tuning.

Cost Calculations for Human and LLM-Based Workflows

Another dimension of research design competitiveness is cost. Its various facets include the required learning curve for startup, time, and monetary cost. Table 4 previews estimations for the time and monetary cost required to set up a CAP-style research project for 5,000 and 100,000 observations for a single-language project labeling newspaper articles (we provide the detailed calculations below).[11]

The LLM-based research design is all about initial investment. The deployment of LLMs requires specialized hardware (GPUs) for both pre-training and fine-tuning models. Since suitable pre-trained models are freely available, our main concern is fine-tuning. While it is possible to fine-tune smaller models on smaller GPUs available on desktop computers, the process is prohibitively time-consuming. This all but means that researchers need access to some cloud computing platform (commercial or non-commercial); to be able to use dedicated GPUs; and execute code, manage data, and develop models remotely. While we do not believe that these are insurmountable barriers to entry for quantitatively minded social researchers, they might pose significant fixed costs to overcome.

It is important to note that one can only provide ballpark estimations for deployment costs. Concrete expenses could depend on a variety of factors including hardware specifications, code configurations, seniority of coder, or length of input text. We are excluding (sometimes considerable) research management costs for human projects. We also exclude the salary of the PI, who is assumed to be a quantitative social/data scientist.

Table 3. Model Performance for Manifesto Datasets.

Language	Weighted macro F1	Max. Micro F1	Std. dev. Micro F1
English	0.55	0.76	0.27
Norwegian	0.63	0.80	0.3
Spanish	0.61	0.77	0.24

Table 4. Overview of Costs in USD for a Project of 5000 and 100,000 Newspaper Articles.

	Human Labeling Workflow		LLM-Based Labeling Workflow	
Observations	Cost	Days	Cost	Days
5,000	5,800	73	6,501	71
100,000	100,800	1260	6,503	71

Regarding manual coding, we should always expect to train at least two coders, which can take up to 1 month (Bulut & Yildirim, 2019; Fahey et al., 2019). Here, we calculated with a modest estimation of 40 hours for training per person and 20 observations labeled in an hour (both need to take place for two independent coders). For a project of 5,000 observations, 10,000 observations is the final count (for two research assistants), which can be done in 580 hours for a total cost of USD 5,800. A similar project for a larger dataset of 100,000 observations (which would cover 10 front-page articles per 300 days a year for a dataset time frame of around 33 years) would cost USD 100,800. For an overview of the calculation inputs, see Table 5.

A quantitatively minded social scientist/data scientist can master a fine-tuning pipeline in Python in up to 2 months (320 hours). A good quality (double-blind coded) training set of 5,000 can yield decent results for model fine-tuning, which requires 10,000 observations coded (for 5,000 USD) to which the cost of training (800 USD) should be added also in this case (for a total cost 5,800 USD). Both model fine-tuning and labeling of virgin data must be done on GPUs due to computational and runtime requirements. Fine-tuning an LLM takes a few hours to a day with the appropriate infrastructure (NVIDIA A40 and A100 GPUs). We calculated half an hour for using the 5,000-row training data for fine-tuning, with an average hourly cost of USD 6.

With a trained model at hand, the coding process on newly submitted data takes from a few minutes to a few hours (which only requires some fundamental maintenance cost—which we ignore here—besides the computational cost). The cost and runtime of labeling are substantially smaller than that of fine-tuning. Based on the Google Cloud Platform billing information for the Babel project, labeling takes around 3 minutes for 5,000 rows and 21 minutes for 100,000 rows and costs USD 0.15 and USD 1, respectively.

Calculating with 8-h work days, the length of the human project is 73 days for a 5k project and 1,260 days for a 100k project. The length of the machine coding project is the sum of an initial 500 hours, or 63 days (as the training set creation can run parallel to mastering LLM-fine-tuning), up to seven days of setting up VMs and 3–21 minutes of labeling time depending on project size. Based on these calculations, the LLM-based workflow is competitive for the smallest of CAP-style media projects. It is vastly preferable for full-scale projects covering multiple decades' worth of data. The outcome of this rough comparison shows that as projects scale up, an LLM-based research design has a clear competitive advantage in terms of cost and time (and a live system can be deployed for future coding at a minimal cost).

Table 5. Costs for a Project of 5000 and 100,000 Newspaper Articles.

		N = 5k				*N* = 100k			
			Cost (USD)				Cost (USD)		
Method	Task	Time in hrs	Initial Cost/hr	Variable Cost/hr	Total	Time in hours	Initial Cost/hr	Variable Cost/hr	Total
Human coding (for two research assistant-RA)	Training	80	10	-	800	80	10	-	800
	Coding 20 obs/hr for 2 RAs)	500	-	10	5,000	10,000	-	10	100,000
LLMs (data scientist)	Learning curve	320	-	-	-	320	-	-	-
	Training set	500	10	0	5,000	500	10	0	5,000
	Fine-tuning	0.5[a]	3[b]	-	1.5	0.5	3[b]	-	1.5
	Labeling	0.05[c]	-	3[b]	0.15	0.35[c]	-	3[b]	1

[a]Based on runtime metrics recorded during fine-tuning of the CAP Babel models.
[b]Based on Google Cloud Platform's hourly pricing for the n1-standard-1 machine type ($0.05) and the hourly pricing of 1 NVIDIA V100 GPU ($2.48).
[c]based on runtime metrics recorded during labeling using the CAP Babel Google Cloud Platform virtual machines.

Conclusion

Our primary objective in this article was to provide a practical solution to researchers applying the CAP classification task to virgin corpora. To achieve this, we set up a pipeline that, for correctly uploaded datasets, may produce CAP major topic prediction results within an hour. The best model results were realized with LLMs fine-tuned on language-domain training sets (e.g., Dutch language—speech domain). By using multilingual XLM-RoBERTa large language models, the pipeline produced state-of-the-art level outputs for selected pairs of languages and domains (such as media or parliamentary speech).

For 24 cases out of 41, the weighted macro F1 of our language-domain models surpassed 0.75 (and, for 6 language-domain pairs, 0.90). Besides macro F1, for most major topic categories, the distribution of micro F1 scores is also centered around 0.75. These results show that the CAP Babel machine is a viable alternative for human coding in terms of validity at less cost and higher reliability. Despite limited training data, it produced gold standard level performance (a weighted F1 of 0.75) for more than half of the language-domain pairs tested.

The proposed research design has significant possibilities for scaling with new models, languages, and datasets for fine-tuning. In light of promising first results and a test of external validity on Manifesto data, we argue (echoing Grimmer and Stewart (2013, p. 281)) that the proposed design can–over time–replace double-blind human coding for a multitude of comparative classification problems as the new gold standard. However, even until such a determination can be unequivocally made, a discussion of the comparative advantages and drawbacks of LLM-based workflows is in order. Since the bulk of the present article focused on measuring the validity of model outputs and also provided detailed cost estimates, we conclude our analysis by contemplating the reliability of Babel Machine results.

According to Krippendorff (2004, p. 211), "a research procedure is reliable when it responds to the same phenomena in the same way regardless of the circumstances of its implementation." Optimal reliability means receiving the same output (in our case, major topic class) for the same input regardless of the number of times the process is initiated. The CAP Babel Machine eliminates such reliability issues: it always returns the same code for the same observation. It also produces softmax scores that allow for additional confidence analysis of outputs. In short, it provides superior results on this dimension of the research design in comparison to human coding.

In sum, our advice to researchers sitting on the fence regarding the incorporation of LLMs into their research design is to embrace them even if they are only used for validating hand-coded results. This can be done at virtually no cost via the Babel Machine (or at some cost with commercial services). Nevertheless, our current results and those of similar recent studies point to the superiority of LLMs vis-á-vis both human coding and traditional machine learning in terms of validity (F1) while its 100% reliability rate is a theoretical maximum. For those technically inclined, therefore, our advice is to use LLMs as a first choice and resort to human coding for validation or active learning processes focusing on less tractable domains/languages.

Acknowledgments

We acknowledge the research assistance of Zoé Baumgartner, Csaba Molnár, and Richárd Nagy. We would like to thank Christoffer Green-Pedersen, Yannick Léonard, Amnon Cavari and Laura Chaqués-Bonafont for kindly sharing data for our experiments.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Governments of Czechia, Hungary, Poland, and Slovakia through Visegrad Grants from the International Visegrad Fund (22310057), Mesterséges Intelligencia Nemzeti Laboratórium (RRF-2.3.1-21-2022-00004), Horizon 2020 Framework Programme (101008468, SLICES-RI), the V-Shift Momentum Project of the Hungarian Academy of Sciences and the HUN-REN Cloud (https://science-cloud.hu/).

Data Availability Statement

Replication material is available at: https://figshare.com/s/8e3e9d1ae22c07869d6d. The CAP Babel Machine service is available at: https://babel.poltextlab.com/. The fine-tuned XLM-RoBERTa models are available at: https://huggingface.co/poltextlab.

Supplemental Material

Supplemental material for this—article is available online.

Notes

1. We are thankful for the anonymous reviewers for their useful and actionable comments. We thank the discussants and participants of the Comparative Agendas Conference (2023), COMPTEXT Conference (2023), and ECPR General Conference (2023) for their suggestions.
2. Models that assign probabilities to word sequences and are trained using at least billions of words.
3. The sources were: Adler and Wilkerson (2015), Alexandrova et al. (2014), Belchior et al. (2019), Borghetto et al. (2019), Breunig and Schnatterer (2019), Chaqués-Bonafont et al. (n.d), Green-Pedersen and Mortensen (2019), Grossman (2019), Guinaudeau (2018), John et al. (2013), McLaughlin (2019), Sebők and Boda (2018), The Policy Agendas Project at the University of Texas at Austin (2017), Walgrave et al. (n.d), Wolbrecht (2016).
4. We followed the domain classification established on the official CAP website's data repository page (https://www.comparativeagendas.net/datasets_codebooks). We added the category "social media" and we separated the original "Parliamentary & Legislative" category into "parliamentary speech" and "legislative" (short for legislative documents). Similarly, we separated the "Prime Minister & Executive" into "executive orders" and "executive speech". Regarding CAP major topic names we refer to "Government operations" as "Government" in the text.
5. At the time of writing, the options for open-source and multilingual models were limited to few widely used and reliable models: XLM, BERT (multilingual), XLM-RoBERTa, and for neural translation M2M and MBart. The larger open-sourced LLMs which are predominantly focused on generative text output are not truly multilingual. Meta's Llama 2's pre-training data is 89.7% English, and the second largest language is German with a 0.17% share (Touvron et al., 2023). For other new LLMs no information is shared on pre-training data composition (the most recent example is the Mistral LLM).
6. We tokenized the texts using the XLM-RoBERTa tokenizer from the Transformers library. It is important to note that tokens are not equal to words, as when a word is not in the vocabulary the tokenizer will create sub tokens. e.g., GPU will be GP, U.
7. The model checkpoint can be accessed via this link: https://huggingface.co/xlm-roberta-large. For the Transformers library documentation see: https://huggingface.co/docs/transformers/index.
8. The softmax function converts the model output into a probability distribution of N outcomes, where N is the number of known categories. One alternative measure of uncertainty, k-fold cross validation of input data, would result in a k-fold increase of run times and cost.

9. To check the robustness of our models, we hyperparameter-tuned the Spanish speech model on a 4*5 parameter space (batch sizes: 4, 8, 16, 32; learning rates: 1e-3, 1e-5, 5e-5, 1e-6, 5e-6). The best-performing model achieved 47% accuracy, which is very similar to the Spanish speech model in the table. Based on this, we expect that the parameters used for fine-tuning our models were appropriate.
10. A possible reason for near-perfect model performance on the Hungarian-budget domain is that budget line items are very repetitive across time. Hungarian budget line items are very repetitive over time (such "Salaries for Defense Ministry personnel"), which results in a lot of duplication or quasi-duplication. This means out-of-sample test data is linguistically very similar to the data used for fine-tuning the model.
11. Such calculations can only provide a ballpark number as concrete costs may vary depending on various factors.

References

Adler, E. S., & Wilkerson, J. (2015). *Congressional bills project: 1989-2014*. Comparative Agendas Project. https://www.comparativeagendas.net/us

Albaugh, Q., Soroka, S., Joly, J., Loewen, P., Sevenans, J., & Walgrave, S. (2014). Comparing and combining machine learning and dictionary-based approaches to topic coding. *7th annual Comparative Agendas Project (CAP) conference* (pp. 1–18). Comparative Agendas Project.

Alexandrova, P., Carammia, M., Princen, S., & Timmermans, A. (2014). *Measuring the European council agenda: Introducing a new approach and dataset*. Comparative Agendas Project. https://www.comparativeagendas.net/eu

Barberá, P., Boydstun, A. E., Linn, S., McMahon, R., & Nagler, J. (2021). Automated text classification of news articles: A practical guide. *Political Analysis*, *29*(1), 19–42. https://doi.org/10.1017/pan.2020.8

Barberá, P., Casas, A., Nagler, J., Egan, P. J., Bonneau, R., Jost, J. T., & Tucker, J. A. (2019). Who leads? Who follows? Measuring issue attention and agenda setting by legislators and the mass public using social media data. *American Political Science Review*, *113*(4), 883–901. https://doi.org/10.1017/S0003055419000352

Baumgartner, F. R., Breunig, C., & Grossman, E. (Eds.), (2019). *Comparative policy agendas: Theory, tools, data*. Oxford University Press. https://library.oapen.org/handle/20.500.12657/52225

Baumgartner, F. R., & Jones, B. D. (1991). Agenda dynamics and policy subsystems. *The Journal of Politics*, *53*(4), 1044–1074. https://doi.org/10.2307/2131866

Belchior, A. M., Borghetto, E., & Moury, C. (2019). *Portuguese agendas project*. Comparative Agendas Project. https://www.comparativeagendas.net/portugal

Bevan, S. (2019). Gone fishing: The creation of the comparative agendas project master codebook. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy agendas: Theory, tools, data* (1st ed., pp. 17–34). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0002

Bevan, S., & Jennings, W. (2019). The UK policy agendas project. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy agendas: Theory, tools, data* (1st ed., pp. 176–183). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0020

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., & Liang, P. (2021). *On the opportunities and risks of foundation models*. arXiv:2108.07258. https://doi.org/10.48550/arXiv.2108.07258

Borghetto, E., Carammia, M., & Russo, F. (2019). *Italian agendas project*. Comparative Agendas Project. https://www.comparativeagendas.net/italy

Breeman, G. E., Then, H., Kleinnijenhuis, J., van Atteveldt, W., & Timmermans, A. (2009). Strategies for improving semi-automated topic classification of media and parliamentary documents. *67th annual meeting of the Midwest political science association*, 1–14. Midwest Political Science Association. https://library.wur.nl/WebQuery/wurpubs/385462

Breunig, C., & Schnatterer, T. (2019). Political agendas in Germany. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy Agendas: Theory, tools, data* (1st ed., pp. 97–104). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0010

Bulut, A. T., & Yildirim, T. M. (2019). The Turkish policy agendas project. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy Agendas: Theory, tools, data* (1st ed., pp. 167–175). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0019

Burscher, B., Vliegenthart, R., & De Vreese, C. H. (2015). Using supervised machine learning to code policy issues: Can classifiers generalize across contexts? *The Annals of the American Academy of Political and Social Science*, *659*(1), 122–131. https://doi.org/10.1177/0002716215569441

Chaqués-Bonafont, L., Palau, A. M., & Muñoz, L. M. (nd). *Spanish policy Agendas*. Comparative Agendas Project. https://www.comparativeagendas.net/spain

Collingwood, L., & Wilkerson, J. (2012). Tradeoffs in accuracy and efficiency in supervised learning methods. *Journal of Information Technology & Politics*, *9*(3), 298–318. https://doi.org/10.1080/19331681.2012.669191

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. arXiv:1911.02116, 1–12.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-Training of deep bidirectional transformers for language understanding. arXiv:1810.04805, 4171–4186. https://doi.org/10.48550/arXiv.1810.04805

Dun, L., Soroka, S., & Wlezien, C. (2021). Dictionaries, supervised learning, and media coverage of public policy. *Political Communication*, *38*(1–2), 140–158. https://doi.org/10.1080/10584609.2020.1763529

Fahey, K., Merle, P., Cornacchione, T., & Weissert, C. (2019). Agenda-setting in the Florida legislature. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy agendas: Theory, tools, data* (1st ed., pp. 200–209). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0023

Flaounas, I., Ali, O., Lansdall-Welfare, T., De Bie, T., Mosdell, N., Lewis, J., & Cristianini, N. (2013). Research methods in the age of digital journalism: Massive-scale automated analysis of news-content—topics, style and gender. *Digital Journalism*, *1*(1), 102–116. https://doi.org/10.1080/21670811.2012.714928

Frantzeskakis, N., & Seeberg, H. B. (2023). The legislative agenda in 13 African countries: A comprehensive database. *Legislative Studies Quarterly*, *48*(3), 623–655. https://doi.org/10.1111/lsq.12404

Gava, R., Varone, F., Mach, A., Eichenberger, S., Christe, J., & Chao-Blanco, C. (2017). Interests groups in Parliament: Exploring MPs' interest affiliations (2000-2011). *Swiss Political Science Review*, *23*(1), 77–94. https://doi.org/10.1111/spsr.12224

Green-Pedersen, C., & Mortensen, P. B. (2019). *Danish policy agenda project*. Comparative Agendas Project. https://www.comparativeagendas.net/dk

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, *21*(3), 267–297. https://doi.org/10.1093/pan/mps028

Grossman, E. (2019). *French agendas project*. Comparative Agendas Project. https://www.comparativeagendas.net/france

Guinaudeau, I. (2018). *Documentation on the coding of German manifestos*. Comparative Agendas Project. https://www.comparativeagendas.net/germany

Hemphill, L., Russell, A., & Schöpke, A. (2019). The rhetorical agenda: What twitter tells us about congressional attention. *77th Annual Meeting of the Midwest Political Science Association* (pp. 1–23). Midwest Political Science Association.

Hillard, D., Purpura, S., & Wilkerson, J. (2008). Computer-assisted topic classification for mixed-methods social science research. *Journal of Information Technology & Politics*, *4*(4), 31–46. https://doi.org/10.1080/19331680801975367

John, P., Bertelli, A., Jennings, W., & Bevan, S. (2013). *Policy agendas in British politics*. Comparative Agendas Project. https://www.comparativeagendas.net/uk

Jungblut, J., Kavli, T. M., & Valgermo, J. B. (2023). Different governments, similar agendas? Analyzing more than seven decades of Norwegian policy agendas presented in executive speeches. *Scandinavian Political Studies*, *46*(3), 167–193. https://doi.org/10.1111/1467-9477.12252

Jurka, T. P. (2012). Maxent: An R package for low-memory multinomial logistic regression with support for semi-automated text classification. *The R Journal*, *4*(1), 56–59. https://doi.org/10.32614/rj-2012-007

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models. arXiv:2001.08361. https://arxiv.org/abs/2001.08361

Karan, M., Šnajder, J., Širinić, D., & Glavaš, G. (2016). Analysis of policy agendas: Lessons learned from automatic topic classification of Croatian political texts. In N. Reiter, B. Alex, & K. A. Zervanou (Eds.), *Proceedings of the 10th SIGHUM workshop on language technology for cultural heritage, social sciences, and humanities* (pp. 12–21). Association for Computational Linguistics. https://doi.org/10.18653/v1/W16-2102

Kleinnijenhuis, J., van Atteveldt, W., & Welbers, K. (2013). De herkomst van vertrouwen in de rechtsstaat 1993–2012: Onderzocht via tekstmining van de mediaberichtgeving en analyse van de publieke opinie. *Wetenschappelijk Ondezoeks- en Documentatiecentrum, Ministerie van Veiligheid en Justitiie*. https://www.wodc.nl/onderzoeksdatabase/tekstmining.aspx?cp=44&cs=6796

Krippendorff, K. (2004). *Content analysis: An introduction to its methodology.* Sage Publications.

Kuipers, G., & Timmermans, A. (2021). From wife to presidential partner: The policy agenda of the first lady of the United States. *White House Studies*, *14*(4), 357–381. https://hdl.handle.net/1887/3480037

Laver, M., & Garry, J. (2000). Estimating policy positions from political texts. *American Journal of Political Science*, *44*(3), 619–634. https://doi.org/10.2307/2669268

Lehmann, P., & Zobel, M. (2018). Positions and saliency of immigration in party manifestos: A novel dataset using crowd coding. *European Journal of Political Research*, *57*(4), 1056–1083. https://doi.org/10.1111/1475-6765.12266

Lewandowski, J., Merz, N., & Regel, S. (2020). manifestoR: Access and process data and documents of the manifesto project. https://CRAN.R-project.org/package=manifestoR

Loftis, M. W., & Mortensen, P. B. (2020). Collaborating with the machines: A hybrid method for classifying policy documents. *Policy Studies Journal*, *48*(1), 184–206. https://doi.org/10.1111/psj.12245

Lucas, C., Nielsen, R. A., Roberts, M. E., Stewart, B. M., Storer, A., & Tingley, D. (2015). Computer-assisted text analysis for comparative politics. *Political Analysis*, *23*(2), 254–277. https://doi.org/10.1093/pan/mpu019

Máté, Á., Sebők, M., Łukasz, W., Dariusz, S., & Ádám, F. (2023). Machine Translation as an Underrated Ingredient? Solving Classification Tasks with Large Language Models for Comparative Research. *Computational Communication Research*, *5*(2), 1–34. https://doi.org/10.5117/CCR2023.2.6.MATE

McLaughlin, J. (2019). *Pennsylvania policy database project*. Comparative Agendas Project. https://www.comparativeagendas.net/pennsylvania

McLaughlin, J. P., Wolfgang, P., Leckrone, J. W., Gollob, J., Bossie, J., Jennings, J., & Atherton, M. J. (2010). The Pennsylvania policy database project: A model for comparative analysis. *State Politics and Policy Quarterly*, *10*(3), 320–336. https://doi.org/10.1177/153244001001000306

Merz, N., Regel, S., & Lewandowski, J. (2016). The manifesto corpus: A new resource for research on political parties and quantitative text analysis. *Research & Politics*, *3*(2), 205316801664334. https://doi.org/10.1177/2053168016643346

Mikhaylov, S., Laver, M., & Benoit, K. R. (2012). Coder reliability and misclassification in the human coding of party manifestos. *Political Analysis*, *20*(1), 78–91. https://doi.org/10.1093/pan/mpr047

Miller, B., Linder, F., & Mebane, W. R. (2020). Active learning approaches for labeling text: Review and assessment of the performance of active learning approaches. *Political Analysis*, *28*(4), 532–551. https://doi.org/10.1017/pan.2020.4

Navarretta, C., & Hansen, D. H. (2022). The subject annotations of the Danish parliament corpus (2009-2017)—evaluated with automatic multi-label classification. *Proceedings of the thirteenth language* *resources and evaluation conference* (pp. 1428–1436). European Language Resources Association. https://aclanthology.org/2022.lrec-1.153.pdf

Pires, T., Schlinger, E., & Garrette, D. (2019). How multilingual is Multilingual BERT? arXiv:1906.01502. https://arxiv.org/abs/1906.01502

Purpura, S., & Hillard, D. (2006). Automated classification of congressional legislation. *Proceedings of the 2006 international conference on digital government research* (pp. 219–225). Digital Government Society of North America.

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2006). An automated method of topic-coding legislative speech over time with application to the 105th-108th U. S. Senate. *64th Annual Meeting of the Midwest Political Science Association* (pp. 1–61). Midwest Political Science Association.

Rauh, C. (2018). Validating a sentiment dictionary for German political language—a workbench note. *Journal of Information Technology & Politics*, *15*(4), 319–343. https://doi.org/10.1080/19331681.2018.1485608

Rytting, C. M., Sorensen, T., Argyle, L., Busby, E., Fulda, N., Gubler, J., & Wingate, D. (2023). Towards coding social science datasets with language models. arXiv:2306.02177. https://arxiv.org/abs/2306.02177

Sebők, M., & Boda, Z. (2018). *Hungarian policy agendas project*. Comparative Agendas Project. https://www.comparativeagendas.net/hungary

Sebők, M., & Kacsuk, Z. (2021). The multiclass classification of newspaper articles with machine learning: The hybrid binary snowball approach. *Political Analysis*, *29*(2), 236–249. https://doi.org/10.1017/pan.2020.27

Sebők, M., Kacsuk, Z., & Máté, Á. (2022). The (real) need for a human touch: Testing a human–machine hybrid topic classification workflow on a New York Times corpus. *Quality and Quantity*, *56*(5), 3621–3643. https://doi.org/10.1007/s11135-021-01287-4

The Policy Agendas Project at the University of Texas at Austin. (2017). *Hearings*. Comparative Agendas Project. [dataset].https://www.comparativeagendas.net/us

Timmermans, A., & Breeman, G. (2019). The Dutch policy agendas project. In F. R. Baumgartner, C. Breunig, & E. Grossman (Eds.), *Comparative policy agendas: Theory, tools, data* (1st ed., pp. 129–135). Oxford University Press. https://doi.org/10.1093/oso/9780198835332.003.0014

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., & Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv: 2307.09288. https://arxiv.org/abs/2307.09288

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *NIPS'17: Proceedings of the 31st international conference on neural information processing systems* (pp. 6000–6010). Curran Associates Inc. https://arxiv.org/abs/1706.03762

Volkens, A., Bara, J., & Budge, I. (2009). Data quality in content analysis. The case of the comparative manifestos project. *Historical Social Research*, *34*(1), 234–251. http://www.jstor.org/stable/20762343

Walgrave, S., Joly, J., Hardy, A., Zicha, B., Sevenans, J., & Assche, T. V. (nd). *Belgian agenda-setting project*. Comparative Agendas Project. https://www.comparativeagendas.net/belgium

White, A. R., Nathan, N. L., & Faller, J. K. (2015). What do I need to vote? Bureaucratic discretion and discrimination by local election officials. *American Political Science Review*, *109*(1), 129–142. https://doi.org/10.1017/S0003055414000562

Wilkerson, J. D., & Casas, A. (2017). Large-scale computerized text analysis in political science: Opportunities and challenges. *Annual Review of Political Science*, *20*(1), 529–544. https://doi.org/10.1146/annurev-polisci-052615-025542

Wolbrecht, C. (2016). *American political party platforms: 1948-2008*. Comparative Agendas Project. https://www.comparativeagendas.net/us

Young, L., & Soroka, S. (2012). Affective news: The automated coding of sentiment in political texts. *Political Communication*, *29*(2), 205–231. https://doi.org/10.1080/10584609.2012.671234

Author Biographies

Miklós Sebők (Ph.D) is a Senior Research Fellow at the HUN-REN Centre of Social Sciences in Budapest. He currently serves as the research director of the Hungarian Comparative Agendas Project and the research co-director of the Artificial Intelligence National Lab at HUN-REN CSS, Budapest. His primary research interests are in political economy and legislative studies, and the application of machine learning and quantitative text analysis in these fields.

Ákos Máté studied political economy (Ph.D) and network science. His main research interest is the application of quantitative text analysis and other big data methods in the field of political economy.

Orsolya Ring (Ph.D) serves as research fellow of the Institute for Political Science at the HUN-REN Centre for Social Sciences in Budapest. In her research, she is focusing on the methodological issues of text analysis especially on emotion analysis using Large Language Models in social science within the Artificial Intelligence National Lab at HUN-REN CSS, Budapest.

Viktor Kovács graduated with a degree in Computer Science Engineering, specializing in neural networks and natural language processing. He worked on the diagnostic classification of schizophrenia using state-of-the-art deep learning models. In poltextLAB his current interests are focused on the fine-tuning, deployment, and scalability of large language models across various domains in the social sciences.

Richárd Lehoczki graduated with a degree in Computer Science. As a developer in poltextLAB he primarily works in machine learning, web development, database administration, and DevOps.