Quality & Quantity (2022) 56:3621–3643
https://doi.org/10.1007/s11135-021-01287-4

The (real) need for a human touch: testing a human–machine hybrid topic classification workflow on a New York Times corpus

Miklos Sebők[1] · **Zoltán Kacsuk**[1,2] · **Ákos Máté**[1]

Accepted: 30 November 2021 / Published online: 17 December 2021

Abstract
The classification of the items of ever-increasing textual databases has become an important goal for a number of research groups active in the field of computational social science. Due to the increased amount of text data there is a growing number of use-cases where the initial effort of human classifiers was successfully augmented using supervised machine learning (SML). In this paper, we investigate such a hybrid workflow solution classifying the lead paragraphs of New York Times front-page articles from 1996 to 2006 according to policy topic categories (such as education or defense) of the Comparative Agendas Project (CAP). The SML classification is conducted in multiple rounds and, within each round, we run the SML algorithm on n samples and n times if the given algorithm is non-deterministic (e.g., SVM). If all the SML predictions point towards a single label for a document, then it is classified as such (this approach is also called a "voting ensemble"). In the second step, we explore several scenarios, ranging from using the SML ensemble without human validation to incorporating active learning. Using these scenarios, we can quantify the gains from the various workflow versions. We find that using human coding and validation combined with an ensemble SML hybrid approach can reduce the need for human coding while maintaining very high precision rates and offering a modest to a good level of recall. The modularity of this hybrid workflow allows for various setups to address the idiosyncratic resource bottlenecks that a large-scale text classification project might face.

Keywords Content analysis · Text mining · Supervised machine learning · News classification

✉ Miklos Sebők
sebok.miklos@tk.hu

[1] Centre for Social Sciences, Budapest, Hungary

[2] Hochschule der Medien, Stuttgart, Germany

1 Introduction

With the advent of large-scale computational solutions being more and more readily available in the humanities and social sciences, quantitative political science is rapidly being transformed by the opportunities offered by machine learning solutions to both existing and new problems. Agenda setting in the times of mass use of social media (Barberá et al. 2019), predicting roll call votes in the US (Bonica 2018) or polarization of lawmakers in the UK (Peterson and Spirling 2018), identifying media frames using newspaper articles (Nicholls and Culpepper 2020), or assessing climate change discourse (Farrell 2016) are just a few of the possibilities opened up by advances in machine learning. Moreover, researchers are also exploring how to utilize neural network architectures in order to use images as data for social science research (Williams et al. 2020).

The advances made in computational social science (of which both computational political science and computational communication science are part of) often leveraged cutting edge machine learning approaches as well as making the case why these methods should appeal to the broader social science research community (Theocharis and Jungherr 2020; Wilkerson and Casas 2017). These advances in the adoption of text mining and machine learning solutions in political science have also opened up new possibilities for major international comparative programs like the Comparative Agendas Project (CAP) and the Manifesto Research on Political Representation (MARPOR) undertaking. As Barberá et al. (2019) show, there are several possible approaches to classify news items with varying degrees of success.

As these datasets increase in size, so does the human and computational effort of labeling items. This scaling problem can be managed by using a hybrid workflow that relies on an efficient combination of human and machine coding. There are some recent proposals that rely on such an approach and demonstrated promising results on Danish and Hungarian language corpora (Loftis and Mortensen 2020; Sebők and Kacsuk 2021). The common theme of this strand of the literature is the quest for finding the optimal workload distribution between the costly human labor and the (often imprecise) supervised machine learning (SML) classification.

The goal of this article is to investigate how a hybrid, human–machine workflow generalizes to an English language corpus. We undertake this challenge by classifying articles in a novel New York Times (NYT) corpus according to the CAP coding scheme (Baumgartner et al. 2019), which distinguishes between 21 major topics. We run simulations on this corpus and compare our results to the corresponding, publicly available human-coded dataset (Boydstun 2013). In keeping with the approach of previous studies, we use Support Vector Machine classifiers (SVM) as the baseline algorithm, but we also extend the scope of the comparison by using Naïve Bayes classifiers (NB) as well. In total, we examine seven SML setups in this article: a multi-class NB, comparison of SVM and NB ensemble workflow, and finally, five different SML hybrid workflows using SVM (from no human coding to one incorporating an element of active learning).

The simulations in this paper offer valuable insights into the potential of incorporating human validation between classification rounds—as opposed to only applying validation at the beginning or the end of the classification process—of a so-called Hybrid Binary Snowball (HBS) workflow (for the details see the discussion of Fig. 1). Alongside the simulation results for various workflow setups, we also introduce indicators that allow us to quantify the gains of using such a hybrid approach where the machine learning and human coding elements are combined to potentially halve the human coding work while attaining an

above 90% precision in classification. We contribute to recent research on the gains in efficiency when coders switch to validation of already coded documents from coding "virgin" texts (Loftis and Mortensen, 2020).

This paper provides evidence that a hybrid workflow can be used in large-scale academic projects to address project-specific bottlenecks, save costs and allocate work more efficiently. We believe that the methods demonstrated below showcase a robust and modular approach that can be applied in many text classification contexts (not limited to policy topic classification). This modularity applies to the supervised machine learning classifiers, adjusting the ensemble to the computational constraints and accomodating the workflow for the difficulty of the classification task. This paper contributes to the growing literature on hybrid workflows by showing that employing a human—machine division of labor can bring tangible benefits for large-scale text classification projects. Moreover, we examine a range of possible supervised algorithms and provide benchmark results for projects using the CAP media labels for categorizing media corpora. The evidence presented in this paper clearly shows the benefits of hybrid workflows and helps social science researchers to make informed ex-ante decisions about structuring large-scale text classification projects.

In what follows, we first cover the various text classification approaches in the political science and communication fields with a special emphasis on the applications and limitations of supervised machine learning solutions. To situate our results, we also highlight the efforts of the CAP research community aimed at automated content analysis. In the methods and data section the hybrid workflow is covered in more detail as well as the reasons for choosing various SML classifiers. The next section shows our simulation results, and the Discussion details the difference between the various SML classifiers and the effectiveness of the hybrid workflow with different levels of human validation. In the Conclusion we discuss the implications of using human validation efficiently in large scale projects and we also highlight some further avenues of research.

2 Literature review

Computer-assisted text analysis has a wide range of tools for analysis, as already detailed in the seminal overview by Grimmer and Stewart (2013). These include using a dictionary to score each document in a corpus, supervised machine learning approaches for classifying documents, and probabilistic topic models (see also Lucas et al. (2015)). A popular approach in this line of research is the use of dictionaries to use word frequency to assign a class to a given document. As Laver and Garry (2000) demonstrate, dictionaries can be used to substitute expert coding of party manifestos. One key advantage of the dictionary-based method is that one can rely on already existing dictionaries and apply them to the appropriate domain. The use of the Lexicoder dictionary to code the sentiment of political communications and economic news demonstrates how out-of-the-box dictionaries can be used effectively (Soroka et al. 2015; Young and Soroka 2012).

However, for text classification tasks, the literature often turns to various supervised machine learning (SML) algorithms. These include, for example, Support Vector Machines, Naïve Bayes classifier, random forests, word embeddings, and various neural network architectures. It is generally agreed that the general SML approach is well suited for classifying large bodies of text given that a training data set of high quality is available (Hillard et al. 2008; Purpura and Hillard 2006). The SVM classifier is often used as a

go-to option as its performance is often superior to the Naïve Bayes, a frequent alternative (Loftis and Mortensen 2020).

A recent comparison between dictionary-based classification and SML forcefully demonstrated that the supervised approach consistently outperforms dictionaries both in terms of accuracy and in terms of precision.[1] Barberá et al. (2019) pit the SentiStrengh (Thelwall et al. 2012), the Lexicoder (Young and Soroka 2012), and an economic dictionary from Hopkins et al. (2017) against regularized logistic regression. According to their findings, the performance of the baseline SML approach hinges on the training sample size, although–as they show–even with as few as 250 observations, the performance of SML already matches the dictionary results (and with 2000 observations, it significantly surpasses it). These results reinforce the already mounting evidence in the quantitative social sciences that SML-based approaches are outperforming the dictionary-based method given the sufficient quality and quantity of training data. It is important to highlight that the two classification methods can be used together to amplify each ' 'solution's strength. Dun et al. (2020) experiments with such a combination and show that one can construct a virtuous loop between the dictionary and SML approaches which results in better dictionary building and better SML results.

The wider political science and social science literature in general is also adopting newer and newer tools, including word embedding based methods, such as GloVe (Pennington et al. 2014) and Word2Vec (Mikolov et al., 2013). In a recent overview, Rodriguez et al. (2021) show that using pre-trained word embeddings in some cases can perform as well as humans in certain tasks.

Regarding the performance of the SML classification across various domains and languages, the social science body of research is considerably thinner. Burscher et al. (2015) examine SML performance on a set of news articles from three different Dutch newspapers and parliamentary questions over 16 years, and they find that the composition of the training set is crucial for classifier performance.

The Comparative Agendas Project (CAP) is one of the premier international scholarly networks in comparative politics, which applies a consistent multi-class policy topic classification scheme. The CAP research project also provides the data and codebook necessary for SML applications in multiple languages, which presents a fertile ground for experiments for teasing out the most efficient way of using human coders in the research workflow.

In an early example of addressing the transition to machine coding in CAP, Hillard et al. (2008) confirms that "for accurately, reliably, and efficiently classifying large numbers of complex individual events, supervised learning systems are currently the best option" (p. 44). Nearly a decade later, these findings are reconfirmed again and again as the comprehensive study of Barberá et al. (2019: 35) also concludes that one should "classify by machine but verify by human[s]". However, the multi-class challenge posed by using the CAP coding scheme also means that even with large training data and using various SML methods, precision results are rarely above 80% in the literature.

Despite the enormous amount of textual data needed to be coded, many of the CAP research teams still rely exclusively on human coding. As the relative cost of computational power steadily decreased, and various ML tools have become available in widely-used programming languages such as R and Python, some country projects of CAP started to

[1] Precision is the fraction of relevant instances among the retrieved instances, while recall is the fraction of the total amount of relevant instances that were actually retrieved.

branch out to automated policy agenda classification (either dictionary-based or a combination of dictionaries and SML, see Albaugh et al., 2013, 2014; Burscher et al., 2015). Nevertheless, most CAP project leaders reported that they did not incorporate ML approaches in their workflow.[2]

While still cheaper than coding an entire corpus by hand, SML classifiers require a training set that might include up to 20% of the entire corpus, and human coding of this portion can also mean sizeable expenses for smaller research teams. There have been some attempts to cut down on costs, either by coding only the leading paragraph for the training set, or implementing a workflow to optimize the human coding and machine classification (Burscher et al. 2015; Loftis and Mortensen 2020; Sebők and Kacsuk 2021). Recent research also emphasizes the gains in efficiency when coders switch to validation of already coded documents from coding "virgin" texts (Loftis and Mortensen 2020). These hybrid approaches also allow for quantifying the gains of using such a human–machine workflow where the SML classification and human coding elements are combined to potentially halve the human coding work while attaining an above 80% precision in classification.

The road paved by previous research is shown in Table 1. Notably, there is no clear 'leader of the pack' approach that dominates the literature. The most recent advances in developing hybrid workflows show that researchers are now focusing on augmenting the SML approaches to break through the ~ 0.8 precision ceiling. The above overview of the key literature shows that our test of the proposed hybrid workflow addresses a gap in the literature and should contribute to our further understanding of the trade-offs and benefits of various human–machine hybrid workflows.

3 Data and methods

3.1 The classification workflow

In this article, we follow the hybrid workflow detailed in Sebők and Kacsuk (2021) which is designed for imbalanced multi-class text classification problems for both single corpus and multi-corpora projects. By default, supervised machine learning methods are hybrid approaches in a sense that the training sample used to "teach" the SML algorithm is labelled by human coders. However, with the below-described hybrid workflow, we emphasize the division of labor between the various human inputs (classifying and validating) and the SML classifier.

Figure 1 presents the outline of this workflow with a few modifications from the above-mentioned article. We focus on the problems researchers might face during coding large bodies of texts, and the workflow variants detailed in the latter half of the article are designed with this production mentality in mind. The implication of this viewpoint is that we put an emphasis on precision (the fraction of relevant instances among the retrieved instances) and recall (the fraction of the total amount of relevant instances that were actually retrieved) separately rather than some aggregated metrics, such as the commonly used F1 score (the harmonic mean of the precision and recall scores), which might cover up insufficient precision values.

[2] For select quotes from CAP project leads see Loftis and Mortensen (2020, 187).

Table 1 Overview and comparison of previous SML results

Article	Methods used	No. of categories	Evaluation metrics	Corpus
Laver and Garry (2000)	Dictionary	19	Pearson corr. against human coding (r = 0.7 +)	Manifesto
Purpura et al. (2006)	SVM	20 and 226*	Accuracy (total acc. = 82.2)	US congressional bills
Hillard et al. (2008)	SVM, MaxEnt, Boostexter, NB, Ensemble	20 and 226 *	Accuracy** (SVM = 88.7 and 81; MaxEnt = 86.5 and 78.3; Boostexter = 85.6 and 73.6; NB = 81.4 and 71.9; Ensemble = 89 and 81	US congressional bills
Young and Soroka (2012)	Dictionary	2	Explained variance (max $R^2 = 15.6\%$)	New York Times
Albaugh, Quinn et al. (2014)	SVM, GLMNet, MaxEnt, Dictionary	20*	Precision (SVM = 0.39, GLMNet = 0.41, MaxEnt = 0.42, Dict. = 0.38), Recall (SVM = 0.29, GLMNet = 0.348, MaxEnt = 0.31, Dict = 0.51)	De Standaard (Flemish newspaper)
Soroka et al. (2015)	Dictionary	2	–	New York Times, Washington Post
Burscher et al. (2015)	Passive Aggressive learning algorithm	20*	F1 score = 0.71	Media corpus (3 most read Dutch newspapers)
Barberá et al. (2019)	Dictionary, Logistic regression	2	Accuracy (Dict. = 60.5, LR = 71), Precision (Dict = 37.5, LR = 71.3)	New York Times
Loftis and Mortensen (2020)	Naïve Bayes	20 and 226 *	Accuracy (total acc. = 71)	Titles of US Billls
Dun et al. (2020)	Dictionary, Random Forest	4	Precision (Dict. = 57.9, RF = 65.6), Recall (Dict. = 0.58, RF = 0.63)	Media corpus (17 highest circulation US newspapers)

*CAP major topics and subtopics, respectively, ** Numbers are for the two label sizes, respectively

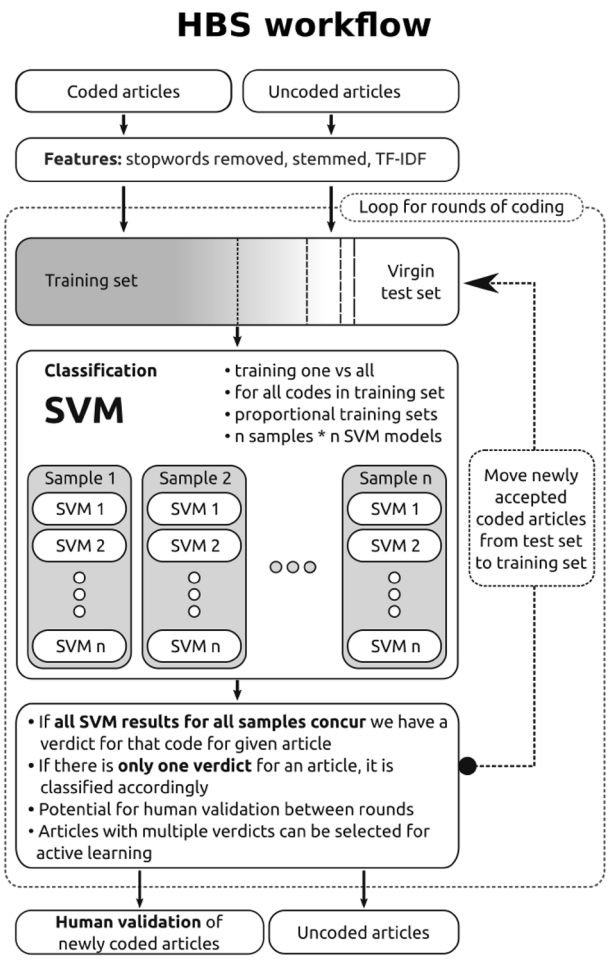

Fig. 1 Overview of the hybrid binary snowball workflow. Adapted from Sebők and Kacsuk 2021

The reason precision is prioritized so highly in this approach is that, as we explain below, we have to make sure that the unvalidated results from the machine classification can be reliably used in future coding and research. Furthermore, as will be discussed in the second half of the article, each project faces idiosyncratic research constraints, be it a lack of funding for human coding or a lack of sufficiently large computational infrastructure. The flexibility of the hybrid workflow model allows for accommodating setups that are tailored to the specific limitations of the project at hand. It is important to note that there will be a portion of the newly classifiable texts that will not lend themselves to classification by machine learning and will have to be coded by human experts. This is in line with the goal of moving as much work as possible to machine-based classification, with only the remaining items requiring human attention.

In order to obtain comparable results over language domains and SML classifiers, we mirror the workflow detailed in Sebők and Kacsuk (2021)and also apply the text pre-processing steps that are standard in the literature, such as removing stopwords, stemming,

and term frequency—inverse document frequency (tf-idf) weighting (Denny and Spirling 2018).

The SML classification is conducted in multiple rounds, and within each round, we run the SML algorithm on n samples and n times if the given algorithm is non-deterministic (e.g., SVM).[3] If all the SML results agree on a single label for a document, then it is classified as such (this approach is also called a "voting ensemble"). We repeat this ensemble voting in each round.[4] The advantages of classifying corpora in this way is that human validation can be incorporated between rounds (either for validation or for active learning where ambiguous cases are referred to the human "oracles"). For the purposes of the simulations below, we use three-fold cross-validation, for which we separate the data into three parts. Two are used as the equivalent of the human-coded initial training set, and the last third is the analog of newly classifiable documents.

Another important feature of this approach is the binary decomposition which simplifies the multi-class classification problem into a series of binary ones by classifying each CAP major topic code against the rest of the codes. This is commonly called a "one versus all" approach. As an example, this means that the SML classifier first looks at the document and decides if it is "macroeconomics" (which is the first category of the CAP major topics) or "other". This logic is then iterated over the whole corpus and for all major topics. This setup allows for a simplified human validation—as opposed to full multi-class coding—where the coders only have to make a binary decision ("correct" or "incorrect") rather than choose from the whole range of available major topics. The advantages of binary classification have been long discussed in the machine learning community and during our measurements we also found this transformation to be crucial to working on imbalanced multi-class text classification tasks (see also Allwein et al. 2000).

The third element in the workflow loop consists of an iterative expansion of the training set (termed snowballing in Sebők and Kacsuk 2021) where items are added to the training set without human validation either automatically or—when human validation of samples is involved—above a certain sample precision threshold. Relying on adding non-validated units to the training set involves an element of trade-off between costs saved and precision lost, with the former significantly outweighing the latter, as we demonstrate below.

The iterative expansion of the training set is an oft-used technique to enhance the SML performance (Olsson 2009); however, it is important to note, that traditionally the selected elements are classified with their correct labels by the queried oracle (most often a human agent). In snowballing, the emphasis is on employing non-validated elements in the process of training set expansion. This does not mean that active learning cannot be incorporated into the workflow, as we will go on to demonstrate in the second half of the discussion, but it is nevertheless important to emphasize that snowballing is a very different approach of

[3] We use a bagging-type classifying ensemble of support vector machines, that incorporates an undersampling of negatives to boost the recall values for smaller classes (Kumar and Gopal 2010; Lango and Stefanowski 2018). The positives of each class are trained against all other classes as its negatives in a one versus all setup as explained above. Where the ratio of positives to negatives in the training set is less than ten percent the program samples the negatives to provide a more balanced training set. To offset the random element introduced in the classification process due to the undersampling of negatives and as presented by the non-deterministic reaching of equilibria by SVMs, we have multiple rounds of both sampling and SVM training in our ensemble.

[4] The optimal number of SVM training rounds to be performed for each sample will be discussed in relation to our simulation results below. Texts with multiple verdicts are disregarded, however, we will demonstrate how we can make use of these elements for active learning in our simulations below.

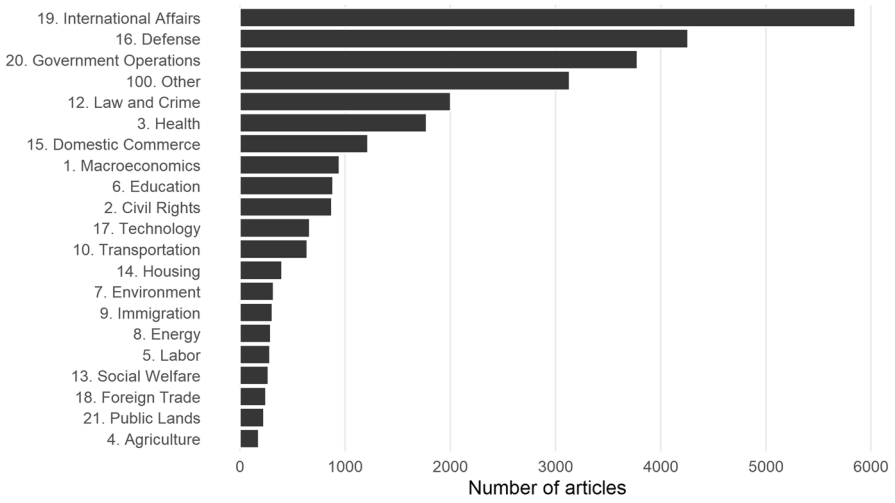

Fig. 2 Topic distribution of the NYT sample

training set expansion, one that is aimed at cost-performance optimization (as opposed to identifying and dealing with items which are hard to classify in active learning).

3.2 Data

Our aim in this study is to extend the external validity of previous studies on hybrid multi-class classification in comparative politics and communication, as well as to test the impact of different setups on research results. We undertake these tasks by using a pre-existing hand-coded dataset of New York Times articles (Barberá et al. 2019; Boydstun 2013; Soroka et al. 2015) and adding the text of the lead paragraphs to these articles via the New York Times API.[5] This data allows us to evaluate the performance of the hybrid workflow against the gold standard of human coding by using parts of this data without the human labels and then validating the machine labeled results against the human-coded data.[6] Moreover, this approach also makes it possible to simulate how human coders would provide validation between each SML classification round.

[5] While these studies obtained the corpus using the Lexis-Nexis database, we opted to download our corpus via the freely available New York Times API (https://developer.nytimes.com/apis). The exact API calls and replication code for the corpus can be accessed at the Harvard Dataverse repository: https://doi.org/10.7910/DVN/I24CYV . For the sake of convenience, we have also provided a preprocessed and anonymized version of our corpus, in which all texts are tokenized, stemmed, changed to lower case, stripped of numbers and punctuation, and tokens are alphabetically sorted; basically providing a summary of the word distributions for each text in accordance with NYTimes copyright and API terms. As a result, the methods working under the "bag of words" assumption will work with this preprocessed corpus, however, the actual content of the lead paragraphs cannot be discerned. The original metadata of the Comparative Agendas Project dataset is available here: https://www.comparativeagendas.net/files/nyt-front-page-complete-data. More information about the data collection, variables, and topic coding of the original dataset can be found in their codebook. The original dataset covers 31,034 observations spanning the years 1996–2006.

[6] For a thorough discussion on the importance and pitfalls of validation, see Song et al. (2020).

Table 2 Comparison of various supervised classifiers

	Naïve Bayes		Random Forest		LASSO regression		Support Vector Machine	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
1. Macroeconomics	0.67	0.73	0.67	0.70	0.80	0.60	0.78	0.75
2. Civil Rights	0.53	0.49	0.20	0.80	0.84	0.36	0.64	0.50
3. Health	0.75	0.76	0.74	0.65	0.76	0.76	0.78	0.75
4. Agriculture	0.58	0.52	0.02	1.00	0.74	0.32	0.81	0.49
5. Labor	0.38	0.30	0.02	0.17	0.46	0.25	0.48	0.45
6. Education	0.76	0.70	0.85	0.67	0.78	0.78	0.81	0.80
7. Environment	0.58	0.52	0.12	0.80	0.70	0.24	0.69	0.51
8. Energy	0.71	0.66	0.29	0.88	0.88	0.6	0.80	0.67
9. Immigration	0.56	0.47	0.32	0.77	0.88	0.52	0.72	0.61
10. Transportation	0.67	0.65	0.36	0.77	0.72	0.44	0.71	0.64
12. Law and Crime	0.65	0.63	0.55	0.61	0.66	0.57	0.66	0.69
13. Social Welfare	0.51	0.45	0.42	0.89	0.69	0.62	0.77	0.63
14. Housing	0.38	0.32	0.18	0.78	0.83	0.32	0.52	0.39
15. Domestic Commerce	0.58	0.59	0.5	0.55	0.62	0.46	0.63	0.61
16. Defense	0.74	0.74	0.82	0.69	0.74	0.76	0.76	0.78
17. Technology	0.61	0.56	0.22	0.69	0.66	0.32	0.65	0.54
18. Foreign Trade	0.46	0.47	0.04	1.00	0.73	0.42	0.71	0.56
19. International Affairs	0.78	0.74	0.75	0.70	0.59	0.8	0.78	0.78
20. Government Operations	0.78	0.81	0.86	0.69	0.76	0.84	0.79	0.85
21. Public Lands	0.38	0.32	0.06	0.60	0.64	0.19	0.51	0.34
100. Other	0.56	0.69	0.69	0.50	0.58	0.67	0.63	0.74
Total	0.6	0.58	0.41	0.71	0.72	0.52	0.7	0.62

Our new corpus contains 28 548 documents from 1996 to 2006. The difference in sample size between our API accessed data, and the coded NYT data is due to the reduced availability of matched text data through the NYT API (instances where captions or title or date did not create a match between the two datasets). For the main descriptive statistics and major code distribution of our new dataset, see Fig. 2. The four most populous categories in the sample are "International Affairs", "Defense", "Government Operations", and "Other". The "Other" category is a catch-all for all CAP media categories above code 21.

4 Baseline simulations

In order to investigate the external validity of the hybrid workflow highlighted in the introduction, we apply this approach to the English language lead paragraphs of the New York Times. Our aim is to classify them into the correct one (the double-blind hand-coded label) of the 21 CAP major categories. To establish a baseline, we classified the documents in the corpus using the commonly used supervised methods (see Table 1.): Naïve Bayes, Random Forest, LASSO regression, Support Vector Machine.[7] The aim of this is to have a baseline

[7] With the exception of the Random Forest, all models were run using threefold cross validation.

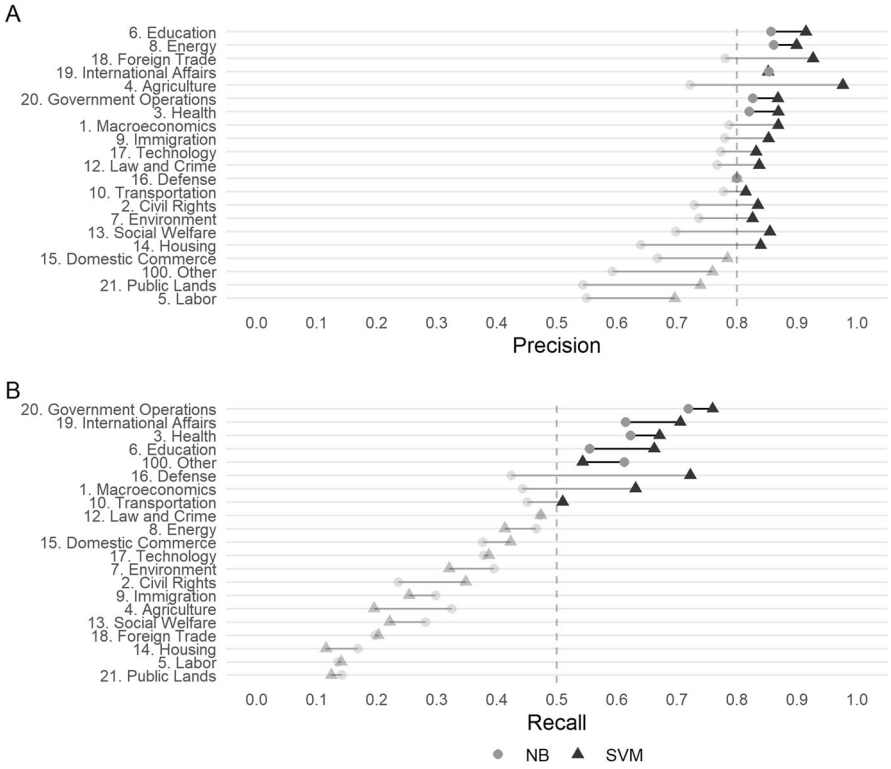

Fig. 3 Scores for "machine only" NB and "machine only" SVM workflows. Dotted lines separate cases where at least one classifier reached 0.8 precision or 0.5 recall

without using the multi-round, hybrid ensemble setup that closely resembles prior research designs and results, as discussed above.

The category-wise precision and recall scores for this setup are shown in Table 2. Given the focus of this paper, we highlight the disaggregated results, as opposed to composite indicators such as the F1 score, as they convey more detailed information. The focus on precision results is also a necessity as—based on our experiments—the snowballing element relies on achieving a 75% or higher precision to work, and if the SML classification is not able to reach this threshold, the snowballing will result in an undesirable drop in the performance of the SML classifier.

In line with what one could have expected based on the reviewed literature, neither the precision nor the recall exceeds 80% for these non-hybrid baseline setups. The poorest performer is the Random Forest classifier (both in terms of precision and recall), while the LASSO regression and SVM perform relatively well. Overall, the SVM produces the best results due to its second-best precision and best recall. However, Table 2 shows that the performance of each classifier varies greatly across categories, with only a handful of

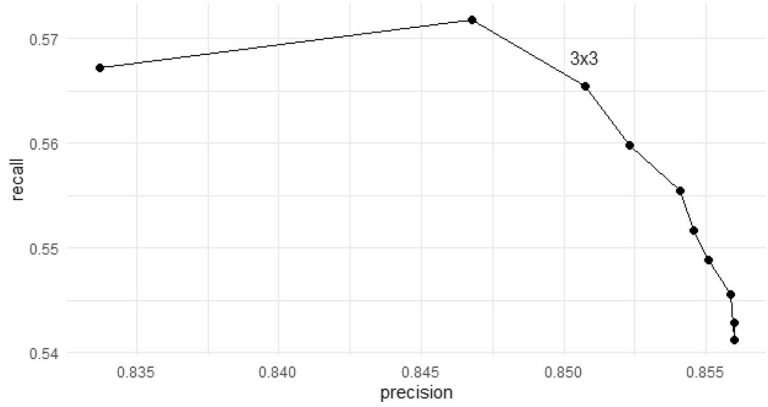

Fig. 4 Flipped precision and recall curve. The 3×3 setup is the first which is above 85% precision. Also note the scales of the y and x axis are not starting from 0. The figure is zoomed in to show the changes between each setup, although in substantive terms the differences are quite marginal

instances of above 80% precision or recall (out of the 84 category predictions, only 11 and 10 categories, respectively).

Now we move to the hybrid setups, comparing the Naïve Bayes and SVM performance. These two approaches were selected based on Tables 1 and 2: the NB classifier is a widely used benchmark in the literature that performed relatively well in the baseline modeling; the SVM offers balanced performance, and it serves as a possible improvement over the NB approach. Three rounds of classification are applied with a 7 sample×7 model SML ensemble. For each round of classification, we employ automatic snowballing for all newly classified elements, regardless of any precision threshold. Figure 3 compares the SVM, and Naïve Bayes-based SML approaches.

When checking the results against the baseline multi-class NB classifier, the improvement is apparent for both the NB and SVM-based classification processes. For the Naïve Bayes classifier, the advantages of the hybrid workflow are apparent: there are six categories with above 80% precision, and out of these three (Energy, Education, and International Affairs) are above 85%, with 86.1%, 85.7%, 85.3% precision scores respectively. The gain in precision is paid for in recall, however, as panel B on Fig. 3 demonstrates. With the hybrid workflow, only four categories are above 60% recall, and only Government Operations reach 72%. This, however, is an expected result. The simple multi-class Naïve Base classifier assigns a class to every single text in the corpus, whereas the classifying ensemble in the hybrid workflow only labels texts that are unanimously agreed on by all models in the ensemble.

Furthermore, as our literature review suggested, the SVM-based workflow outperformed the one building on Naïve Bayes classifiers. Matched against the NB setup (leaving everything unchanged but the SML algorithm), the SVM-based workflow performs better in 19 categories out of the 21 based on precision and 13 times for recall. The gains in precision are quite large, and the hybrid workflow using the 7 samples×7 models setup and SVM classifiers is able to produce precision values above 85% in 10 categories and above 80% in 16 categories. This performance is a significant improvement compared to previous SML approaches to classifying news documents into the 21 CAP major categories.

In light of the results in Fig. 3, the SVM classifier-based workflow clearly performs better than the Naïve Bayes classifier-based one. Therefore, for the remainder of this article, we focus on measurements relying on SVM. In order to determine the optimal number of samples and training rounds to be used in our further simulations, we run a comparison of the classifying ensembles from 1 sample × 1 model all the way to 10 samples × 10 models. The precision and recall trade-offs are shown in Fig. 4.

Based on these simulations, we settled on a 3 × 3 setup for our further work as it was the first to achieve precision over 85%. This choice is, of course, specific to our desired results, as the recall difference between different setups is below 0.3 percentage points. While our selection criteria depend on the rather artificial 85% cut-off point, this can vary from project to project as each has its idiosyncratic resource constraints (and, therefore, different tolerance for lower recall). Finally, we note that the larger the ensemble, the more computation time it will take to get the necessary results. This means that any incremental increase in precision for recall might not be a worthy trade-off if the underlying infrastructure is costly or slow.[8]

4.1 Experimental simulations focusing on optimal human annotation

4.1.1 How to use human annotators?

One of the most important questions for hybrid workflow solutions is how to allocate human working hours in the most optimal fashion. Here, we once more emphasize the problem of the "human touch": is it best to use human annotators as coders or as validators, and in what proportion. As it was demonstrated by Loftis and Mortensen (2020), coders reliably performed better when validating rather than labeling virgin documents. Following this work, we also contacted a national CAP research group on their experience. Based on discussions with expert coders who supervise coding teams using a hybrid workflow and moving annotators from coding to validating tasks cut the hours needed in half.[9]

It is important to emphasize that these results are dependent on the expertise of the annotator. Another important insight is that beginner-level annotators are more likely to agree with the SML classification during validation, so there is a risk that false positives slip through the validation process. Having said that, the speed-up of the process is in part due to the binary classification problems that validators have to make in each round: instead of deciding if a certain article belongs to one of 21 categories, they only have to decide if the label assigned by the SML is correct or not. Our conclusion from the above results and this qualitative information is that the same allotment of annotator-hours yields better results if split between the creation of the training set and validation. But the application of the "human touch" also leads to different results when used as in-process validation vs. post-process validation. On the one hand, introducing human validation between classification rounds contributes to training set expansion and, even in the worst case of no new true positives, identified false positives could be held back from being added to the training set. On the other hand, all things kept equal, this would lead to an increase in the

[8] Based on our experiments, which were carried out in the cloud setup described in Pintye et al. (2019), a round of 3 × 3 SVM ensemble took around 2.7 h, while a round of 10 × 10 SVM ensemble took around 30 h to run.

[9] The inter-coder reliability for coding various corpora in the project varies but it is generally in the 70–85% range (which also depends on the amount of training and experience a coder pair has).

validation burden for the overall process or sap resources dedicated to ex-post validation (and would, therefore, lead to reducing the size of feasible projects). In the rest of the article, we examine the opportunities opened up by the inclusion of human validation between classification steps and introducing a number of indicators for measuring the cost–benefit performance of the way we make use of human labor. In short, we will be able to provide a more substantial answer to the question of optimizing the allocation of human working hours in our process.

4.1.2 Using alternative setups to gauge annotator efficiency

In this section, we compare four additional workflow setups (beyond the core SML setup discussed previously), all of which include sampling-based human validation steps between classification rounds. We will refer to the above core SML setup as the "machine only" baseline since it only relies on automated between-rounds training set expansion and no actual validation. That is, the newly added items to the training set could be either true positives or false positives, and gold-standard human validation is only applied ex-post. Figure 5 provides a summary overview of both the elements shared among all versions and the settings specific to each of the different setups. The employed classifying machine learning solution is the same for each setup, the already described SVM, and the one-vs-all binary decomposition-based ensemble (three samples times three models). Furthermore, to better capture the effect of the different workflow setups, we conducted five rounds of iterative classification and training set expansion.

Whereas the “machine only” setup, as in the measurements before, employs automatic intra-corpus snowballing, adding all newly classified texts to the training set for the next round of classification, the four other setups incorporate an element of human validation between each round of classification. For human validation, random samples are drawn from each class based on the number of classified texts. For classes with less than a hundred classified texts, all are selected. For classes with a higher number of classified elements, sample size is determined based on binomial (representing correctly and incorrectly classified texts) sampling rules so that the validation results allow for a $\pm 5\%$ confidence interval with 95% confidence.

The four setups incorporating human validation built on each other in the following manner. The first version, titled “basic” relies on only adding identified true positives from the validation sample to the training set during the first two rounds; and then allowing for snowballing, that is also adding non-validated non-sample elements to the training set, from round three onward for major topics where the validation sample precision reaches at least 75% (the threshold we use as the standard inter-coder reliability for human coders).

The “enhanced basic” version also takes advantage of the additional information gained about identified false positives through the sample validation process. Once a document has been identified as having been falsely classified as belonging to some class, it will no longer be included in the test set when checking for that given class. Rather it will be included in the training set as a known negative for that class.

Third, the “stepwise balanced” setup is motivated by our goal of better optimizing the allocation of human labor in the process. Depending on the training and test set, some classes might reach very high levels of precision, even starting from the first round of classification. It makes no sense to keep validating samples for these classes, as opposed to snowballing them into the training set straight away. In this setup, we start out with the possibility of snowballing enabled from the very first round, but we employ a conservative

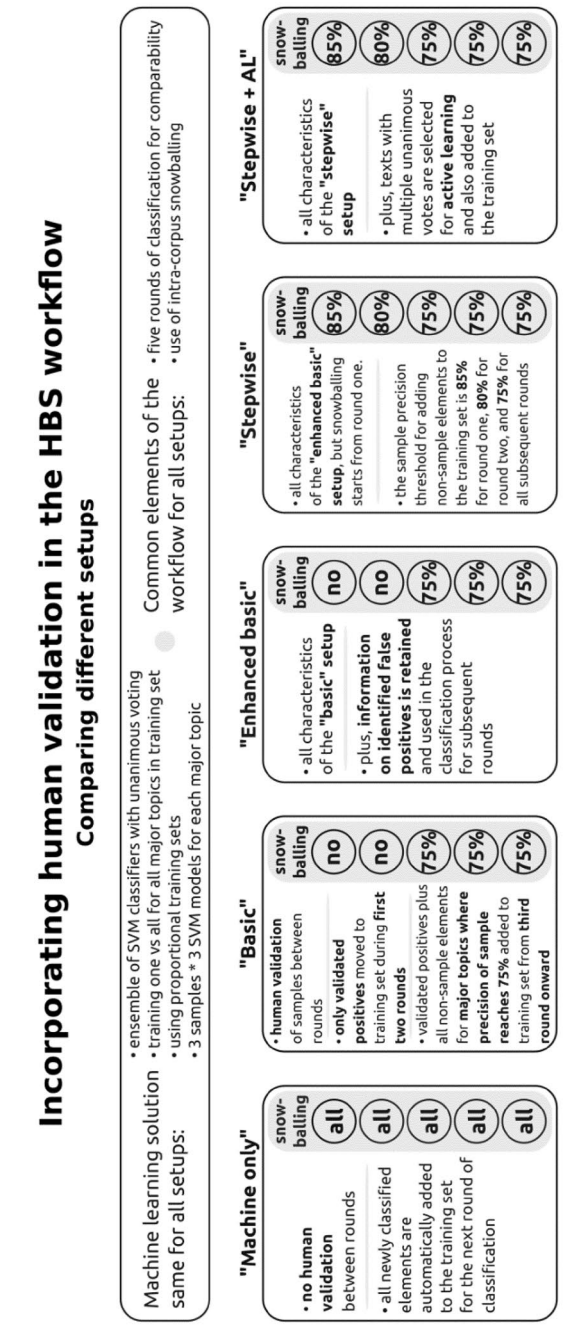

Fig. 5 Overview of the hybrid workflow setups

stepwise approach to lowering the precision threshold for intra-corpus snowballing: 85% in the first round, then 80% in the next round, and finally apply the previously used 75% baseline from round three onward.

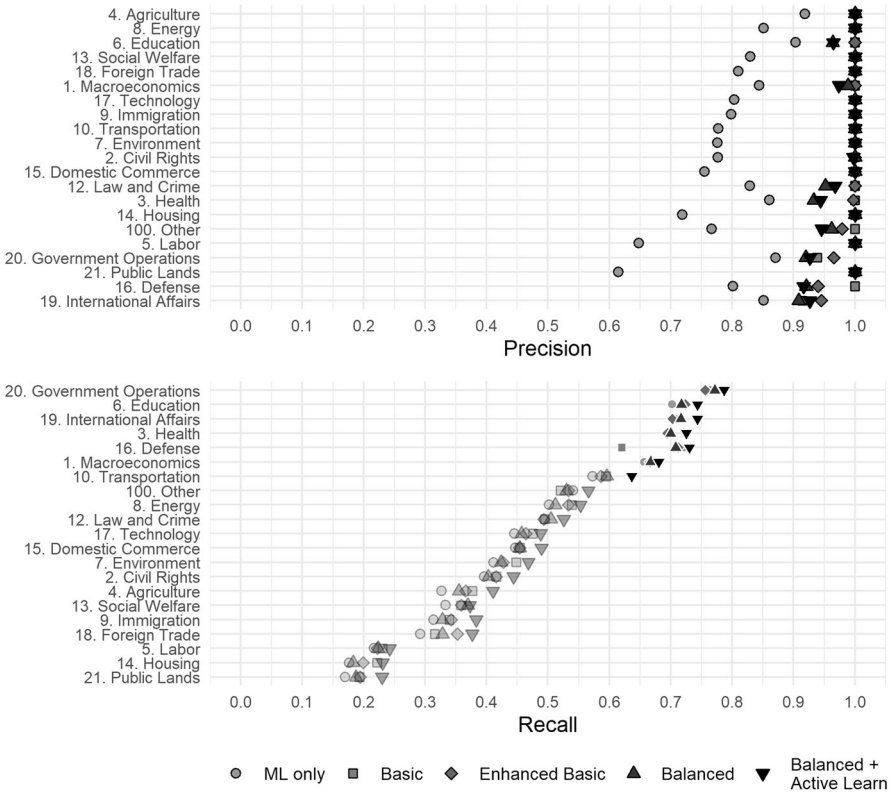

Fig. 6 Overview of machine only and hybrid workflow results

Finally, the "stepwise balanced plus active learning" setup adds an active learning component to deal with documents that are singled out by our classifying ensemble as being highly characteristic of multiple (most often two, sometimes three) classes at the same time. This is active learning in two senses. On the one hand, the elements to be queried are selected based on their obvious elusiveness for our classifying ensemble to handle, as they are the elements that are assigned multiple verdicts (these items have so far been disregarded in all of the previous setups in line with the description of our ensemble above). On the other hand, this is true active learning in the sense that these elements are not validated for the assigned multiple classes but are rather definitively classified by human experts.

4.2 An evaluation of alternative setups: precision

Figure 6 offers a summary overview of the precision and recall results for the five setups. While precision values are dominated by the results for the "basic" and "enhanced basic" setups, the "balanced stepwise plus active learning" produces the highest recall values for all major topics.

To understand these results in the context of what is happening in the five setups, let us turn to the discussion of precision first. The detailed numbers in Table 3 confirm that

Table 3 Overview of precision results for the different setups

Major topic	Machine only	Basic	Enhanced basic	Stepwise balanced	Stepwise balanced + AL
1. Macroeconomics	84.4	100	100	98.9	97.4
2. Civil Rights	77.7	100	100	100	99.8
3. Health	86	100	99.8	93.3	94.4
4. Agriculture	91.9	100	100	100	100
5. Labor	64.8	100	100	100	100
6. Education	90.3	100	100	96.4	96.5
7. Environment	77.6	100	100	100	100
8. Energy	85.1	100	100	100	100
9. Immigration	79.8	100	100	100	100
10. Transportation	77.7	100	100	100	100
12. Law and Crime	82.9	100	100	95.2	96.7
13. Social Welfare	82.9	100	100	100	100
14. Housing	71.9	100	100	100	100
15. Domestic Commerce	75.5	100	100	100	100
16. Defense	80.1	100	94	92.1	91.7
17. Technology	80.3	100	100	100	100
18. Foreign Trade	81	100	100	100	100
19. International Affairs	85.1	91.4	94.6	90.9	92.6
20. Government Operations	87	93.8	96.6	92	92.7
21. Public Lands	61.5	100	100	100	100
100. Other	76.7	100	98	96.2	94.6
Total	82.6	96.8	96.9	93.8	94.2

indeed the precision values for the "basic" and the "enhanced basic" setups are the highest, with the majority of major topic codes standing at a 100% precision rate. This, however, is due to almost all accepted classified elements undergoing human validation with very little actual snowballing taking place in these two setups. The "balanced" and "balanced + active learning" setups, where snowballing is more dominant, perform similarly well, with only 6 major topics below 95% precision and all of the topics above 92%. Compared to the "machine only" baseline, the hybrid workflow shows remarkable improvements across all specifications in terms of precision.

Adding an element of active learning to the classification rounds in the fourth setup not only increases the total precision of our results but also has a clear net benefit on all three of our cost–benefit indicators, as can be seen in Table 5 (with the added cost of the oracle function performed by the human experts on the articles selected for active learning—we return to this issue below). A careful analysis of our results, however, points towards an unexpected possible result, namely that the articles selected for active learning should not be added to the training set. Examining Table 3 again, we can see that some major topic classes (Macroeconomics, Civil Rights, Defense, and Other) do suffer a slight drop in precision.

Table 4 Overview of recall results for the different setups

Major topic	Machine only	Basic	Enhanced basic	Stepwise balanced	Stepwise balanced + AL
1. Macroeconomics	65.7	67.2	66.6	66.8	68
2. Civil Rights	39.6	41.6	41.6	40.3	44.4
3. Health	69.4	69.6	69.5	70	72.6
4. Agriculture	32.7	37.7	36.6	35.5	41
5. Labor	21.7	23.1	22.3	22.3	24.2
6. Education	70.3	71.9	72.4	71.7	74.3
7. Environment	41.1	44.8	42.7	42.4	46.8
8. Energy	50.2	53.9	53.4	51.2	55.3
9. Immigration	31.4	33.9	34.3	32.8	38.3
10. Transportation	57.3	59.4	58.7	59.6	63.6
12. Law and Crime	49.4	49.5	49.3	50.4	52.5
13. Social Welfare	33.3	35.9	35.9	37	37.3
14. Housing	17.6	22.2	19.9	18.3	23
15. Domestic Commerce	44.7	45.4	45.4	45.4	49
16. Defense	72.1	62	71.2	70.8	73.1
17. Technology	44.6	47.6	46.4	45.7	48.8
18. Foreign Trade	29.2	31.6	35.2	32.8	37.6
19. International Affairs	70.6	70.4	70.3	71.6	74.3
20. Government Operations	75.8	76.4	75.6	77.1	78.7
21. Public Lands	17	19.3	19.5	18.8	23
100. Other	54.1	52.1	53.3	53	56.6
Total	61.4	60.4	61.6	62	64.7

4.3 An evaluation of alternative setups: recall

The recall by major topic category is much more heterogeneous than the precision, as Fig. 6 and Table 4 show. The differences between the “machine only” and the various hybrid setups is not as pronounced as it was the case for the precision results. There are 7 categories above 60% recall in the “stepwise balanced + AL” setup and 6 categories in the middle of the road "stepwise balanced" setup. Compared to the precision, the recall (both total and by topic) results are significantly lower. Nevertheless, these values still represent considerable savings in human coding for a text classification project, given the high precision of the workflow. Furthermore, we can see in Table 4 that the total recall score increases by 2.7 percentage points because of activating the active learning component. Adding 436 correctly classified articles to our set of newly classified articles ceteris paribus should increase our total recall score by 4.6 percentage points for an average unclassified corpus set size of 9516. This seems to indicate that while adding the correctly classified active learning elements does indeed have a gross positive effect on all our total indicators, the total change covers up a drop in performance for the results if we were to exclude the active learning elements.

The most likely explanation for this phenomenon lies in the way the text of the articles selected for active learning is characteristic of more than one major topic. For example, an

Table 5 Snowballing gains and validation costs across the hybrid setups

Indicators	Machine only	Basic	Enhanced basic	Stepwise balanced	Stepwise balanced + AL
Total snowballing gain	–	1035	1366	2891	2887
Net snowballing gain	–	843	1176	2498	2788
Total validation cost	–	7522	5677	4229	3965
Total active learning cost/gain	–	–	–	–	436

article on the privatization of hospitals would be hard to classify even for human experts choosing between "macroeconomics" and "health" based on the emphases and slight nuances of the given text. By adding these types of articles to the training set, we are, in fact, in a way lowering the quality of classification. Therefore, going forward, we would recommend still selecting out elements with multiple verdicts for human consideration. But the aim here would not be to add them to the training set per se, only to the final set of newly classified elements at the end of all classification rounds. This would, of course, change the meaning of this component: no learning takes place, as the information is not added to the training set. The reason we would nevertheless recommend selecting these elements out for human experts to check against the ensemble verdicts is because this type of decision between two (in rare cases three) major topic classes, like validation, is again potentially much easier than full multi-class classification.

4.4 Validation cost and snowballing gain

To better appreciate the costs involved in gaining additional precision, we introduce three indicators (see Table 5). The "total snowballing gain" (or automated training set expansion gain) is the number of all non-validated articles that are classified during the process. The "net snowballing gain" is the number of correctly classified texts among the total. Third, the "total validation cost" is the number of classified articles that were checked during the human validation phases of the classification rounds in each case. Note that "total validation cost" also includes the cost of checking false positives. We can see the drop in "total validation cost" from the "basic" to the "enhanced basic" setup, as the retention and use of information on identified false positives helps the classifying ensemble avoid making the same classification errors again.

The drop in precision results as we move from the "enhanced basic" to the "stepwise balanced" setup (see Table 3) is precisely the result of more snowballing happening in the latter workflow. The net snowballing gain increases from 843 documents up to 2788 in the case of the "stepwise balanced + AL" setup. The average starting test set size (the documents to be newly classified) for our simulations is 9516, which means that 29.3% of the test set can be correctly classified automatically and accepted without human validation (while still yielding the exceedingly high precision results in Table 3) using the hybrid workflow coupled with active learning. In line with these gains the validation costs are also decreasing as there are less documents for the human annotators to go over, due to the retention of information on false positives, snowballing starting from round one and the separating out of articles for active learning.

Table 6 Single verdict convergence in the different hybrid workflows

Classification round	Machine only	Basic	Enhanced basic	Stepwise balanced	Stepwise balanced + AL
Round 1	6316	6313	6323	6342	6333
Round 2	581	4405	4019	2248	2501
Round 3	175	3155	2530	629	284
Round 4	85	1393	290	140	76
Round 5	55	1090	99	60	43

It is important to note that proportional validation costs and snowballing gain numbers do not scale linearly (due to the way the sample size is determined based on the binomial distribution), and thus the larger the corpus to classify, the greater the gains that can be reaped.

Furthermore, as indicated above, in the cases where human in-process validation is introduced into the hybrid setup, snowballing becomes tied to the human validation of samples. Therefore, hard to classify—that is, from a precision aspect: underperforming—major topics will not suffer from the problem of automatically adding large portions of false positives into the training set as is the case in the "machine only" setup. This effect is obvious in the way worst-performing major topics (Labor, Housing, and Public Lands) for precision in the "machine only" version all have perfect precision scores in the setups involving human validation.

The introduction of human validation of samples helps to differentiate between the easy and hard to classify major topics. These benefits, of course, will vary from corpus to corpus. But based on this data, our expectation is that in-process validation will lead to savings on both human training set creation and overall (in-process and ex-post) validation work, while at the same time ensuring that texts belonging to problematic major topics are treated with due diligence and are only accepted after human validation.

4.5 The application of hybrid classification with limited computational capacity

So far, we have discussed our results from the viewpoint of the optimal allocation of human labor. Nevertheless, our results also provide useful information for use cases where computational capacity is expensive or limited. In Table 6. we summarize the total number of single verdicts assigned by the classifying ensemble each round for the different setups. It is important to remember that—depending on the setup used—not all single verdicts end up becoming newly classified elements moved to the training set. What this table demonstrates is the speed of convergence with which the workflow reaches a state where the classifying ensemble no longer assigns verdicts to the still unclassified elements in a significant quantity (and for the very last steps, the precision of those results also drops too low for the process to be worth pursuing any further).

In our measurements (based on the setup discussed in the section baseline simulations), the workflow consisted of three rounds, and the acceptability of this cut-off point is further buttressed by the measurements for the current group of simulations. Except for the "basic"

version, all other setups reach their convergence point around the third to fourth round, depending on the cut-off criteria. This is a very important result for projects with a bottleneck in computational capacity. Three rounds of the hybrid workflow using the "stepwise balanced" setup, for example, are sufficient for reaping the benefits of the machine classification process within a hybrid approach.

5 Conclusion

The goals of this article were twofold. First, our aim was to investigate the external validity of the hybrid workflow approach to classify a large body of news documents. Second, we set out to investigate if such a hybrid ensemble setup can be used in a production setup to save on human annotating capacity by placing the human touch where it is most needed.

In our research design we considered four algorithms based on their prevalence in the literature and our initial test results: Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM) and LASSO. We proceeded with NB and SVM due to the fact that (1) NB is widely used in the literature (regardless of its actual performance) and (2) SVM looked to offer better results than LASSO.

As for the first goal, we showed that both the NB and SVM classifiers perform better than the base NB approach on the New York Times corpus. To have the best possible comparison, we replicated the SVM workflow with our English language corpus and found that the workflow yielded comparable results compared to what Sebők and Kacsuk (2021) reports and exceeding the results reported by Loftis and Mortensen (2020). These results also highlight and reinforce the superior performance of the SVM method compared to the Naïve Bayes and the (so far often overlooked) power of using ensemble SML approaches.

Second, we provided evidence that such a hybrid ensemble setup can be used in a production setup to achieve considerable savings in human annotating needs. The four variations on the workflow using different degrees and types of human–machine interaction showed that carefully balancing human and SML efforts can maintain high precision and also save costs. In our simulations the average starting test set size surpassed 9000, and on a corpus of this size we could generate valid machine codes—with no human involvement—for almost 30% of the test set without any meaningful sacrifice to high precision.

This allows research projects to shift resources to address production bottlenecks or complete projects ahead of schedule. Another key advantage of the modular structure is that if an in-process human validation element is incorporated (as we demonstrated above) it provides the best of both worlds: the snowballing process can use categories where the SML classification is very accurate, and human validation can be applied to categories where the SML classifier struggles.

We believe that the methods demonstrated above showcase a robust and modular approach that can be applied in many text classification contexts (not limited to the CAP codebook). Further testing of the workflow might include using the hybrid workflow with different coding schemes and across domains (although one cross-domain application, language, is already underway as we presented our data in comparison to other projects).

While the results show that the hybrid workflow and the SVM voting ensemble perform exceptionally well with unbalanced classification tasks, further robustness checks might look into applying it to different coding frameworks and see how well these results travel across research domains. Similarly important is the fact that adopting this workflow

approach can speed up large text classification projects considerably, and cost savings allow projects to use ML as an efficiency booster in planning and implementation.

Acknowledgements We are grateful to Amber Boydstun, and the participants of EPSA 2021, and ECPR General Conference 2021 for their stimulating feedback. We gratefully acknowledge the role of the ELKH Cloud (https://science-cloud.hu/) infrastructure and the Occopus cloud orchestration tool (https://occopus.readthedocs.io/en/latest/) and express our gratitude to the team of the Laboratory of Parallel and Distributed Systems at ELKH SZTAKI. The project was supported by the Ministry of Innovation and Technology NRDI Office within the framework of the Artificial Intelligence National Laboratory Program. The authors would also like to acknowledge support from the Hungarian Artificial Intelligence National Laboratory.

Funding Open access funding provided by Centre for Social Sciences. Artificial Intelligence National Laboratory of Hungary, 2020, Miklós Sebők

Data availability The data that support the findings of this study are openly available in the Harvard Dataverse: https://doi.org/10.7910/DVN/I24CYV

References

Albaugh, Q., Julie, S., Stuart, S., Peter, J.L.: The automated coding of policy agendas: a dictionary-based approach. In: 6th Annual Comparative Agendas Conference, Antwerp, Belgium (2013)

Albaugh, Q., et al.: Comparing and combining machine learning and dictionary-based approaches to topic coding. In: 7th Annual Comparative Agendas Project (CAP) Conference, 12–14 (2014)

Allwein, E.L., Robert, E.S., Yoram, S.: Reducing multi-class to binary: a unifying approach for margin classifiers. J. Mach. Learn. Res. **1**, 113–141 (2000)

Barberá, P., et al.: Automated text classification of news articles: a practical guide. Polit. Anal. 1–24 (2019)

Baumgartner, F.R., Breunig, C., Grossman, E.: Comparative Policy Agendas: Theory, Tools. Oxford University Press, Data (2019)

Bonica, A.: Inferring roll-call scores from campaign contributions using supervised machine learning. Am. J. Polit. Sci. **62**(4), 830–848 (2018)

Boydstun, A.E.: Making the News: Politics, the Media, and Agenda Setting. University of Chicago Press (2013)

Burscher, B., Vliegenthart, R., De Vreese, C.H.: Using supervised machine learning to code policy issues: can classifiers generalize across contexts? Ann. Am. Acad. Pol. Soc. Sci. **659**(1), 122–131 (2015)

Denny, M.J., Spirling, A.: Text pre-processing for unsupervised learning: why it matters, when it misleads, and what to do about it. Polit. Anal. **26**(2), 168–189 (2018)

Dun, L., Stuart, S., Christopher, W.: Dictionaries, supervised learning, and media coverage of public policy. Polit. Commun. 1–19 (2020)

Farrell, J.: Corporate funding and ideological polarization about climate change. Proc. Natl. Acad. Sci. **113**(1), 92–97 (2016)

Grimmer, J., Stewart, B.M.: Text as data: the promise and pitfalls of automatic content analysis methods for political texts. Polit. Anal. **21**(3), 267–297 (2013)

Hillard, D., Purpura, S., Wilkerson, J.: Computer-assisted topic classification for mixed-methods social science research. J. Inform. Tech. Polit. **4**(4), 31–46 (2008)

Hopkins, D.J., Kim, E., Kim, S.: Does newspaper coverage influence or reflect public perceptions of the economy? Res. Polit. **4**(4), 2053168017737900 (2017)

Kumar, M.A., Madan, G.: A comparison study on multiple binary-class SVM methods for unilabel text categorization. Pattern Recognit. Lett. **31**(11), 1437–1444 (2010)
Lango, M., Jerzy, S.: Multi-class and feature selection extensions of roughly balanced bagging for imbalanced data. J. Intell. Inform. Syst. **50**(1), 97–127 (2018)
Laver, M., John, G.: Estimating policy positions from political texts. Am. J. Polit. Sci. 619–34 (2000)
Loftis, M.W., Mortensen, P.B.: Collaborating with the Machines: a hybrid method for classifying policy documents. Policy Stud. J. **48**(1), 184–206 (2020)
Lucas, C., et al.: Computer-assisted text analysis for comparative politics. Polit. Anal. **23**(2), 254–277 (2015)
Mikolov, T., Ilya, S., Kai, C., Corrado, G.S., Jeff, D.: Distributed representations of words and phrases and their compositionality. In: Burges, C.J.C., Bottou, L. Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Informa- tion Processing Systems. Lake Tahoe, NV: Neural Information Processing Systems, 3111–19 (2013)
Nicholls, T., Culpepper, P.D.: Computational identification of media frames: strengths, weaknesses, and opportunities. Polit. Commun. 1–23 (2020)
Olsson, Fredrik. 2009. "A Literature Survey of Active Machine Learning in the Context of Natural Language Processing."
Pennington, J., Richard, S., Christopher, M.: Glove: Global Vectors for Word Representation. In: Alessandro, M., Bo, P., Walter, D. (eds.). In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha: Association for Computational Linguistics, 1532–43 (2006)
Peterson, A., Spirling, A.: Classification accuracy as a substantive quantity of interest: measuring polarization in westminster systems. Polit. Anal. **26**(1), 120–128 (2018)
Purpura, S., Dustin, H.: Automated classification of congressional legislation. In: Proceedings of the 2006 International Conference on Digital Government Research, 219–225 (2006)
Rodriguez, P., Arthur, S.: Word embeddings: what works, what doesn't, and how to tell the difference for applied research. J. Polit. Ahead of Print (2021)
Sebők, M., Kacsuk, Z.: The multiclass classification of newspaper articles with machine learning: The hybrid binary snowball approach. Polit. Anal. **29**(2), 236–249 (2021)
Song, H., Tolochko, P., Eberl, J.M., Eisele, O., Greussing, E., Heidenreich, T., Lind, F., Galyga, S. Boomgaarden, H.G.: In validations we trust? The impact of imperfect human annotations as a gold standardon the quality of validation of automated content analysis. Polit. Commun. **37**(4), 550–572 (2020)
Soroka, S.N., Stecula, D.A., Wlezien, C.: It's (change in) the (future) economy, stupid: economic indicators, the media, and public opinion. Am. J. Polit. Sci. **59**(2), 457–474 (2015)
Thelwall, M., Buckley, K., Paltoglou, G.: Sentiment strength detection for the social web. J. Am. Soc. Inform. Sci. Technol. **63**(1), 163–173 (2012)
Theocharis, Y., Andreas, J.: Computational social science and the study of political communication. Polit. Commun. 1–22 (2020)
Wilkerson, J., Casas, A.: Large-scale computerized text analysis in political science: opportunities and challenges. Annu. Rev. Polit. Sci. **20**, 529–544 (2017)
Williams, N.W., Andreu, C., Wilkerson, J.D.: Images as Data for Social Science Research: An Introduction to Convolutional Neural Nets for Image Classification. Cambridge University Press (2020)
Young, L., Soroka, S.: Affective news: the automated coding of sentiment in political texts. Polit. Commun. **29**(2), 205–231 (2012)