**A Weakly Supervised and Deep Learning Method
for an Additive Topic Analysis of Large Corpora**

**Abstract**

The collaborative effort of theory-driven content analysis can benefit significantly from the use of topic analysis methods, which allow researchers to add more categories while developing or testing a theory. This additive approach enables the reuse of previous efforts at analysis or even the merging of separate research projects, thereby making these methods more accessible and increasing the discipline’s ability to create and share content analysis capabilities. This paper proposes a weakly supervised topic analysis method that uses both a low-cost unsupervised method to compile a training set and supervised deep learning as an additive and accurate text classification method. We test the validity of the method, specifically its additivity, by comparing the results of the method after adding 200 categories to an initial number of 450. We show that the suggested method provides a foundation for a low-cost solution for large-scale topic analysis.

“Political text as data” has emerged as an important trend in political science and communication studies in recent years. As the volume of and access to political texts continue to grow and computing resources become more available, we see an increasing need for research methods that focus on the systematic extraction of themes, topics, and concepts from large-scale news corpora (Grimmer and King 2011; Grimmer and Stewart 2013; Quinn, Monroe, Colaresi, Crespin, et al. 2010). This paper builds on two recent trends in this field, both of which aim to establish an accessible textual analysis method that can advance empirical research. The first is the use of topic models as an unsupervised topic analysis method to reduce costs by eliminating the need to manually code large amounts of text (Blei 2012; Quinn, Monroe, Colaresi, Crespin, and Radev 2010). The second is the ability to incorporate external variables into the topic model and to explore and test theories by measuring relationships between topics and those variables (Lucas et al. 2015).

In this paper, we suggest that additivity—the ability to add topics to an existing model or even to merge two models—can further contribute to empirical research along these two lines. First, it makes this kind of research more accessible, as researchers can collaborate on projects and identify topics from different domains while reusing existing trained models. Second, it facilitates the testing of theoretical relationships between variables, as it allows for the addition of more topical variables to the theoretical model (e.g., testing whether a relation between variables holds while controlling for other variables). Last, by enabling the analysis of a different and possibly more general corpus, it expands the applicability of the empirical findings. Herein, we show how current methods are limited in these aspects and suggest that using weak supervision, in which the computer learns with “incomplete, inexact or inaccurate supervision” (Zhou 2018, 44), can allow us to merge multiple topic models into a flexible and accessible method for topic analysis.

The outline of this paper is as follows: section 1 reviews current methods and their limitations; section 2 introduces our solution; section 3 describes the compilation of a training set using unsupervised learning; section 4 describes the supervised classifier; section 5 demonstrates and validates our solution’s additivity; section 6 further validates our model; and section 7 presents our conclusion and highlights the advantages of our solution.

# Current Methods of Large-Scale Content Analysis

As a computational content analysis method, topic modeling allows for large-scale analysis that allocates text to multiple categories with minimal human effort. In this context, the computer looks for topics—distributions of words over a vocabulary—based mostly on the frequency and co-occurrence of words in an unsupervised approach without prior coding of text examples. For example, terms such as “game” and “football” are likely to appear more frequently in the topic “sport” compared with terms such as “politics” and “congress” (Blei 2012). Topic models have proven to be a powerful analytical tool that is highly suitable for large corpus analyses with multiple topics of interest (Blei, Ng, and Jordan 2003; Grimmer 2010; Quinn, Monroe, Colaresi, Crespin, and Radev 2010). Recent developments have enhanced the ability to examine theoretical relations between external variables and corpus topics by incorporating covariant variables into a Structural Topic Model (STM). This model has further enhanced topic models’ popularity among computational social science researchers (Roberts et al. 2014).

However, because topic models learn topics inductively instead of being given a list of topics *a priori*, they are sometimes difficult to use when testing a theory involving specific topical variables, which is the common scenario for theory-driven research (Collingwood and Wilkerson 2012; Günther and Quandt 2015; Guo, Vargo, Pan, Ding, et al. 2016; Roberts et al. 2014). In addition, the outcome can be affected by even small variations in processing steps or in the model’s configuration. Therefore, achieving reliable, stable, and reproducible topic models is quite challenging. The problem is aggravated when the corpus is not fixed but continuously expanding, as is the case when collecting and analyzing political speeches, news, and social media during the course of a political campaign (Chuang et al. 2015; Denny and Spirling 2018; Fokkens et al. 2013; Wilkerson and Casas 2017). Topic models are also difficult to evaluate, leading to disagreements between researchers regarding the results of their analyses (Maier et al. 2018). All of this complexity compromises the ability of topic models to produce collaborative and replicable scientific results. Some of these limitations could be resolved if it were possible to add topics to an existing topic model. Unfortunately, there is no simple method for adding topics to an existing topic model (Blei 2012; Schwartz and Ungar 2015).

These limitations may drive researchers to use topic models while exploring a corpus and building a theory, and then use other, more appropriate methods to identify a given list of categories. One such method is dictionary analysis, in which a set of terms is searched for in the text in order to identify the corresponding predefined categories (Burscher, Vliegenthart, and De Vreese 2015; Soroka, Young, and Balmas 2015). Dictionaries are explicit, transparent, and additive. However, creating a valid dictionary is very costly, and adding categories to an existing dictionary may entail even higher costs, as all other categories should first be considered to prevent contradictions (Quinn, Monroe, Colaresi, Crespin, and Radev 2010). In addition, the accuracy of dictionary analysis may be compromised by the choice of terms, and in general, the method tends to suffer from low recall scores (Guggenheim, Jang, Bae, and Neuman 2015; Guo, Vargo, Pan, Ding, and Ishwar 2016). Recent methods have succeeded in reducing the subjective bias that may accompany the manual selection of words, which improves recall, but these approaches further increase start-up costs for creating a dictionary (King, Lam, and Roberts 2017).

Supervised learning, in which the computer learns the weight of each term and considers additional features, such as contextual information, usually results in more accurate classifications than dictionary analysis (Cambria and White 2014; Grimmer and Stewart 2013). It also facilitates the creation of new categories by simply adding labeled text examples to the training set. As such, this method seems to be the best choice for a text classification designed to accurately identify predefined categories that also provides a more reliable, stable, and reproducible way to update the list of categories.

Despite its advantages, studies in the social sciences usually use supervised learning only to identify a small number of categories because of the high cost of identifying each one (Burscher, Odijk, Vliegenthart, de Rijke, et al. 2014; Quinn, Monroe, Colaresi, Crespin, and Radev 2010). In some cases, supervised learning is used merely as a filtering mechanism, and the actual in-depth analysis is performed manually (Nardulli, Althaus, and Hayes 2015). Therefore, even though supervised learning seems to be a natural choice for theory-driven research, its high cost limits its use by social scientists, especially when the tested theory involves more than a few variables.

# Weak Supervision as an Additive Alternative

To solve this problem, we suggest using a weakly supervised method, which reduces manual labor by splitting the training process into two phases. The methods involve first applying a low-cost labeling method to raw data, which minimizes human labor while creating a training set with labels that are useful despite being incomplete or not fully accurate. The training set is used to train a regular supervised or a semi-supervised learning method in order to create a deductive predictive method (Hernández-González, Inza, and Lozano 2016; Zhou 2018). In this way, these methods can reduce the cost of human labor, thus leveraging very large training sets, while providing performance on par with fully supervised learning methods (Hoffmann, Zhang, Ling, Zettlemoyer, et al. 2011). Researchers have also demonstrated how weak and manual annotations can be combined to improve models’ performances even further, thereby creating new paths for collaborative research initiatives (Deriu et al. 2017).

Our approach applies unsupervised learning to a large volume of news articles to compile a training set that is then used to train a separate supervised classifier. It is true that other low-cost methods can be used as alternatives for human coding, such as crowdsourcing (Dehghani, Zamani, Severyn, Kamps, et al. 2017; Rudkowsky et al. 2018). However, our method provides a better use of the available resources, so that projects with more funding can use crowdsourcing to create a training set, while those with more constrained funding can take advantage of access to experts to verify and interpret the outcomes of the unsupervised learning method. We believe one of the reasons for the popularity of topic models in the computational social sciences, and specifically in communication studies, is that many social scientists have more access to experts than they do to funding. Additionally, the specifics of a particular research project can make crowdsourcing less attractive. In a pilot study we performed with a group of six undergraduate coders, it took approximately three months of manual labor to compile a dataset of 10,000 labeled sentences with reasonable inter-coder reliability for less than twenty categories. In a case such as ours, which was likely to entail a larger number of categories, the scale of the coding labor required made crowdsourcing infeasible.

We therefore used unsupervised learning as the first step of our weakly supervised topic analysis method. More concretely, we used topic models as the unsupervised method. Our unique contribution here is the transformation of the output of the topic models into a labeled training set (as described in section 3.3), which renders the use of a weakly supervised solution practical. Using first topic modeling and then supervised learning allowed us to add categories to the training set and to train a supervised classifier to identify existing and new topics without having to retrain and relabel the original topic models (see demonstration in section 5).

# Training Set Compilation

Our solution is composed of two main phases: first, compiling a training set, and then, training a supervised learning classifier. The training-set compilation phase consists of the following steps: (1) collect a corpus of texts (news articles in our case) that belong to a single general subject (e.g., crime, sports); (2) train a topic model at the article level for each corpus; (3) convert topics from the article level to the sentence level; (4) create clusters of sentences based on topic association scores; (5) manually label the clusters; and (6) add the labeled sentences to the training set (see Figure 1 for a schematic overview of the process). In the following, we describe the process in detail, while illustrating it with our example case of a large-scale topic analysis of news articles.

## Collecting Articles for Single-Subject Corpora

We envision a common scenario in which a researcher collects multiple corpora, each relevant to a single general subject (an area of interest) that the researcher wants to divide into more specific categories (e.g., breaking down politics into subcategories of elections, policy, and political campaigns). In addition, we found it technically preferable to train a topic model on a collection of news articles relevant to a single general subject, because such a corpus makes it easier to identify and label topics. To demonstrate the training-set compilation method, we collected articles from the LexisNexis archive, from January 1995 to March 2017, starting with a list of approximately 700 news sources (see the Supplementary Materials). For each general subject, we identified the names of substantially similar newspaper sections (e.g., economy, markets, and finance) based on the collaborative judgment of three experts. We then collected all of the articles found in these sections, without any filtering.

|  |
| --- |
| Figure 1 Scheme of Process |
| C:\Users\Owner\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Rapid Labeling - Page 2 (1).png |
| *Note: Process overview (section numbers in parentheses).* |

## Training a Topic Model at the Article Level for Each Corpus

Before training each topic model, we performed standard preprocessing on each corpus: cleaning; lemmatization; and the removal of punctuation, stop words, common and rare terms, and short texts (for a thorough explanation of these steps, see, for example, Jacobi, van Atteveldt, and Welbers 2016). We then estimated the number of topics based on the size of the corpus (generally 25 to 100 topics). Finally, we trained several Latent Dirichlet Allocation (LDA) topic models (Blei, Ng, and Jordan 2003) until the human coders were satisfied with the results at the labeling step (as described in section 3.5).[[1]](#footnote-2)

## Converting Topics from the Article Level to the Sentence Level

Mixed-membership topic models such as LDA or STM have a useful advantage—their fit for analyzing news articles, since those articles are more likely to contain multiple topics compared to other texts. However, this feature also creates a challenge, as labeling and validating such topic models by reading entire articles is difficult when a researcher cannot exactly identify the section of the article that expresses a specific topic (Maier et al. 2018).

For example, in our demonstration we trained a topic model on a corpus with ‘crime’ as the general subject. When we conducted an article-level analysis of the distribution of topics for a given article entitled “Police: Man arrested in Waterloo police chase sold heroin, crack cocaine,” the two topics with the highest percentages were topic #5 (22.5%) and topic #32 (18.6%). Because the percentages were quite similar, it was difficult to conclusively identify a single main topic for the entire article or clearly identify which section of the article discussed each topic.

Compared to articles, sentences tend to be more focused and hence associated with fewer topics. This makes them much easier to label manually and to use in training a supervised algorithm (Leetaru and Schrodt 2013). However, we must consider that two sentences with similar content can have different meanings, depending on the context of the article, among other parameters. We thus began our analysis at the article level and then moved to the sentence level before we labeled topics. This allowed us to use the rich contextual information at the article level to train the topic model before moving to the sentence level.

Next, we calculated a “topic association score” representing the level of association between sentence *s* and topic *k*. For each topic, the topic association score considers both the broader context of the distribution of topics at the article level and the specific content of the distribution of each sentence’s words over the vocabulary.

Formally, the topic model results in a distribution of topics (*Θd*) for each document *d*, a probability of topic *k* occurring in document *d* (*θk,d*), and a probability of word *w* occurring in topic *k* (*φk,w*). For each sentence *s*, we calculated a topic association score (*TAk,s*) using equation (1). For each topic *k* in the distribution of topics in the document (*Θd*), we multiplied the proportion of topic *k* in document *d* (*θd,k*) by the sum of the values of the corresponding *phi* for each word *w* in the sentence (*φk,w*):

$$\left(1\right) TA\_{k,s}= θ\_{k,d}\*\sum\_{w in s}^{}φ\_{k,w}$$

This results in better differentiation between topics at the sentence level because, instead of a single distribution of topics constant throughout the entire article, each sentence receives different topic association scores based on its specific content (see the follow-up example in section 3.5).

## Clustering Sentences Based on Topic Association Scores

The goal of the training-set compilation phase is to replace the manual labeling of individual sentences, which is an extremely labor-intensive task. We achieved this goal by creating clusters—automatically created groups of sentences—that could be labeled collectively. To this end, we shifted our focus from sentences to clusters, with each cluster corresponding to a topic in the topic model. Instead of taking into account the topic association scores assigned to specific sentences, we were interested in collecting sentences with the highest scores for each topic. We first computed the topic association scores for sentences from the entire corpus. Then, for each topic, we extracted all sentences with a standardized topic association score above two (that is, the top 5% from all sentences), which we used as a minimal threshold for creating sentence clusters. These groupings were then reviewed by the human experts.

## Labeling the Clusters Manually

Human experts played three roles during the training-set compilation phase. First, they judged whether the topic model resulted in “good enough” clusters in terms of clarity and coherence. If not, we reconfigured and retrained the topic model. Once the clustering was considered to be good enough (usually within the first or second attempt), the human experts inferred a label for each cluster by manually reviewing a random sample of sentences. To ensure that the sentences were read in context, we provided the experts with the entire article and title for each sentence. To ensure that the sentence clusters were coherent, we asked the experts to establish an exact threshold for each cluster. This unusual use of human coding was done by arranging the collected sentences in five groups based on their standardize topic association score (2–2.5, 2.5–3, 3–3.5, 3.5–4 and above 4). We then asked the experts to indicate the exact threshold for each topic that would provide a coherent cluster. We required the label and threshold to correspond to each other, since a more general label might lead to choosing a lower threshold, which would include more sentences. For example, consider a case where sentences with the highest topic association score for a given topic are all related to US-Russia relations, but that lower association scores also include sentences related to US-Mexico relations. The human experts were responsible for deciding whether to choose a higher threshold and a narrower label, such as the topic “US-Russia relations,” or to choose a lower threshold and a broader label, such as “US foreign affairs.” This process sometimes required several discussions and iterations until the experts agreed on both the label and threshold.

We now turn back to the example of the news article presented in section 3.3, describing a drug dealer in Waterloo who caused a car accident while fleeing from police. The original topic model we trained operated at the article level with two main topics. Yet, the goal of the training-set compilation phase was to identify texts that were more strongly associated with each topic. When we shifted our focus from the article level to the sentence level, the picture became clear. Sentences involving drivers and vehicles received higher scores on the first topic (#5), while sentences involving drugs received higher scores on the second topic (#32—see Table 1). After reviewing a sample (N=~100) of sentences with high topic association scores for each topic collected from the entire corpus, the human experts assigned the label “Crime—Drivers & Vehicles” to topic #5 and “Drugs” to topic #32.

The manually inferred label was then propagated to all sentences within each cluster. Therefore, unlike traditional methods of manual labeling done with supervised learning, the human experts only reviewed a small fraction of each group of sentences, but the label they inferred was assigned to a much larger group of similar sentences.

At this point, we separated the training-set compilation phase from the training of the supervised learning classifier by imposing two rules. First, we focused our attention on a sentence’s binary association with a category, rather than its actual topic association score (in other words, was the sentence related to the category or not, regardless of whether the sentence was tagged as related to more than one category). This decision made it easier for us to compile a training set of labeled sentences. Second, some words, such as stop words, were removed during preprocessing, and were therefore not given a *phi* value by the topic model and did not contribute to their sentence’s topic association score. However, the clustering and manual label inference were performed using the original sentences, including all words, so that the supervised classification method would be able to take them all into account.

## Adding the Labeled Sentences to the Training Set

|  |
| --- |
| Table 1 Labeling Categories by Reviewing Sentences |
| ***Sentence Text*** | ***Topic Association Scores*** |
| ***Topic #5*** | ***Topic #32*** |
| “He allegedly refused to stop, and intentionally crashed into an unmarked Sheriff's vehicle, causing damage and a hand injury to a deputy.” | **1.64** | 0.03 |
| “McCullough caused damage to the field with the vehicle, and became stuck in mud.” | **1.12** | 0.01 |
| “Seneca County Sheriff's deputies announced additional charges Thursday for a Rochester man allegedly connected to selling illicit drugs in the area.” | 0.34 | **1.88** |
| “McCullough was charged with two counts of third-degree criminal sale of a controlled substance, two felony counts of third-degree criminal possession of a controlled substance, two counts of sale of an imitation controlled substance.” | 0.12 | **2.08** |
| *Note: Example of labeling categories using topic association scores for sentences.* |

We aggregated the labeled sentences into a single training set. In cases where the label of a topic learned by one topic model overlapped with the label from another topic model, we merged both groups into a single group of sentences with one label.

The purpose of the entire process is to train a supervised classifier, and therefore validation should focus on the supervised classifier, while the training set is assumed to contain noise. Nevertheless, in a previous study we evaluated the correctness of the training-set compilation phase by manually reviewing labeled sentences. This evaluation validated the clustering method and the labels given to clusters, and, as a by-product, helped to train our human coders and to fine-tune the process. We therefore recommend that researchers who are interested in applying the process conduct this evaluation, which we describe in more detail in Online Appendix 1.

# Designing and Training a Deep Learning Sentence Classifier

We now turn to the second phase of our weakly supervised method, in which we used the compiled training-set to train a supervised deep learning classifier. A deep learning model usually outperforms classical learning models, as it can learn how to efficiently represent raw data using its hidden layers (Lai, Xu, Liu, and Zhao 2015; dos Santos and Gatti 2014). Unfortunately, deep learning models also usually depend on large amounts of data, sometimes millions of labeled examples (LeCunn, Bengio, and Hinton 2015). This is likely one of the most significant barriers to using deep learning in the computational social sciences, especially when the goal is to identify large number of categories. Yet it is also where we gain the most benefit from the low-cost, unsupervised compilation of the labeled training set. Our design may not be optimal (many other designs can be used as alternative methods of supervised learning), but it provides a valid demonstration of a sufficient method. In the interest of concision, we provide only a brief description of the classifier. (For detailed explanations, see Online Appendix 2. We recommend that researchers who are new to the field of deep learning review this appendix before reading the next section).

## Preprocessing Sentences

Because deep learning models automatically learn how to represent raw data, the preprocessing of text input varies somewhat from classical machine learning techniques. Instead of removing stop words or symbols from the text (Lai, Xu, Liu, and Zhao 2015), we used only the Stanford CoreNLP tokenization tool (Manning et al. 2014) and converted the tokens to lower case. Finally, we removed sentences with fewer than five tokens, assuming they did not contain enough information regarding the relevant category.

## Model Architecture

In our architecture, sentences are represented by a fixed length vector. To allow the model to analyze the complete sentence, we chose a length of 100 words (including punctuation marks), which covers more than 99% of the cases (based on a sample of 10 million sentences). Shorter sentences are padded with zeros at the beginning, which the model ignores. The model’s input layer then embeds the words of these fixed-length sentences into a vector representation, based on GloVe pre-trained vectors (Pennington, Socher, and Manning 2014).[[2]](#footnote-3)

We added a long short-term memory (LSTM) layer to allow the model to learn from sequential information (word order) and from multiword patterns (Bengio, Courville, and Vincent 2012; Hochreiter and Schmidhuber 1997; Lai, Xu, Liu, and Zhao 2015). The LSTM layer was configured to contain 100 memory units so that an entire sentence could be stored in memory simultaneously. To reduce the risk of overfitting the training set, we added a dropout regularization method, configured with a rate of 20% for the input and the recurrent features of the LSTM layer (Gal and Ghahramani 2016; Srivastava, Hinton, Krizhevsky, Sutskever, et al. 2014).

We experimented with different architectures to classify sentences based solely on the text but did not achieve a reasonable level of accuracy. This finding was consistent with our understanding of discourse, whereby the same sentence may have different meanings in different contexts. As a simple solution, we added the article’s title as a contextual input to the model and duplicated the embedding and LSTM.

We concatenated the output of the two LSTM layers into a 200-length vector. The vector was fed into a fully connected network with a number of modules equivalent to the number of categories plus thirty, with a “ReLU” activation function (Krizhevsky, Sutskever, and Hinton 2012; Nair and Hinton 2010). This layer was connected to the output layer with the same number of modules as the number of categories.

Even though a sentence is usually more focused than an entire article, it still may refer to more than one category, especially when the categories are not mutually exclusive. In fact, sentences in the political domain commonly contain multiple topics (consider, for example, sentences from a political debate on public health spending).

We therefore designed the model’s output layer to predict a multilabel classification, such that multiple categories may be predicted for each sentence. To this end, the layer minimized a weighted binary cross entropy loss with a sigmoid activation function (Kurata, Xiang, and Zhou 2016; Nam, Kim, Loza Mencía, Gurevych, et al. 2014). This loss function creates a multilabel classification by separately providing the probability for each category to be true. All categories with a greater than 50% probability of being true were marked as identified. In the end, we used the Adam optimizer to minimize the loss function (Kingma and Ba 2014).

## Training the Sentence Classifier

Once the choice of layers and the individual layer sizes (number of modules) were set, we tuned the hyperparameters. To reduce the risk of overfitting, which can occur during the selection of the best hyperparameters, we split the sentence-level data into three sets: training, validation, and test. We trained the model with different hyperparameters using the training set, chose the best configuration based on the accuracy calculated on the validation set, and tested the accuracy of the final model using the test set (Grimmer and Stewart 2013). We also halted training before any degradation of performance on the validation set, after two epochs (training-cycles) (Srivastava, Hinton, Krizhevsky, Sutskever, and Salakhutdinov 2014).

# Adding Topics Iteratively

One of the advantages of supervised learning is the ability to add more categories to the training set by adding text examples labeled with new categories. Typically, a researcher will simply add manually labeled text examples. Alternatively, the researcher can conduct additional iterations using our method: collect an additional corpus with a general subject, train a topic model, convert its outputs to clusters of sentences, infer a label for each cluster, and add the sentences with the new labels to the training set.

## Illustrating Additivity

An interesting option is to decompose one of the existing categories into more specific ones (see Figure 2). For example, we trained a version of our supervised classifier using sixteen different corpora, as described in section 3. This version (henceforth referred to as Version 15) was trained to identify 450 categories. One of the categories was guns and gun control in the United States, which we extracted from corpora collected using the names of newspaper sections, such as politics and crime. For the sake of this illustration, we wanted to decompose this category into more focused categories, such as attitudes towards gun control and use of guns in the United States. We used the trained supervised classifier to identify news articles in the category of guns and gun control in the United States (e.g., all articles in which this category was identified by more than 10% at the article level). By doing so, we created a new corpus that was focused on gun control in the United States. The ability to create more nuanced categories can enable further empirical research on this topic.

|  |
| --- |
| Figure 2 Adding Topics to the Model by Decomposing Existing Categories |
| Rapid Labeling - Page 3 |
| *Note: Overview of the process of adding topics to the model (section numbers in parentheses).* |

To demonstrate the addition of more nuanced contextually relevant categories, we have repeated this process and decomposed two additional categories, “elections & primary campaigns” and “conflicts,” which we consider likely to be relevant to our topic of gun control. We have also collected another corpus from the opinions sections of various newspapers in order to add more diverse perspectives on potentially relevant political issues. After training a topic model for each corpus and running the rest of the training-set compilation method, we added the resulting labeled sentences to our training set. Combined with the illustration of the original training set used for Version 15, we analyzed 20 corpora containing approximately 30 million articles, resulting in a training set containing about 100 million sentences labeled with a total of 651 topics (see Table 3). We used this training set to train a new supervised classifier (referred to as Version 16).

The entire training-set compilation phase, including the addition of the additional 201 categories (from a training set of 100 million sentences), required approximately 400 work hours by human experts who were tasked with inferring a label and setting a threshold for each cluster. As we have seen in our pilot study mentioned above, the same amount of work required to directly label sentences resulted in a dataset of only 10,000 labeled sentences for less than 20 categories.

## Testing Additivity

To test the additivity of our model, we performed reliability tests at the sentence and article levels. In both tests, we compared the classifications made by the two versions of the model: Version 15, with 450 categories, and Version 16, with 651 categories.

For the first test, we compared the classifications made by the two versions on a reserved test set of 5.99 million sentences, sampled from the compiled training set. As we do not have a gold standard based on human coding that would have allowed an external verification of accuracy, we treated the two versions as two coders and tested their inter-coder reliability. We did not expect complete agreement, since any addition of categories to the model could affect the model’s classification of other related categories. Our expectation was that the agreement between the two versions would be sufficiently high as to indicate the stability of the model despite the addition of new categories.

|  |
| --- |
| Table 3 Collected Corpora |
| ***General subject (“Context”)*** | ***Articles*** | ***Topics*** | ***Labeled Sentences*** |
| Economy | 11,002,527 | 75 | 17,066,574 |
| Education | 281,716 | 50 | 710,898 |
| Elections & Primary Campaigns\* | 300,205 | 50 | 1,144,320 |
| Energy & Natural Resources | 100,435 | 50 | 291,640 |
| Guns & Gun Control in the US\* | 25,707 | 25 | 94,396 |
| Health | 381,093 | 50 | 2,494,971 |
| Immigration | 13,767 | 25 | 28,384 |
| International | 4,433,328 | 75 | 14,647,669 |
| Legal, Crimes, and Police | 949,554 | 50 | 16,639,412 |
| Mideast & The Arab World | 107,031 | 75 | 1,608,840 |
| National Elections & Political Conflicts\* | 2,129,710 | 100 | 8,975,413 |
| National Security | 190,136 | 50 | 917,377 |
| Opinion | 5,000,000 | 100 | 1,222,735 |
| Politics | 953,437 | 50 | 24,121,219 |
| Science | 113,954 | 50 | 551,606 |
| Sport | 5,000,000 | 75 | 1,761,938 |
| Technology | 200,303 | 75 | 3,956,669 |
| Tourism | 688,952 | 50 | 5,195,551 |
| Transportation & Vehicles  | 305,683 | 50 | 1,369,054 |
| Weather | 147,198 | 50 | 582,371 |
| *Note: Collected corpora, each with a general subject, were used to train LDA models with the corresponding number of topics.* *\* The general subject was a topic identified by a previous version of the model. Other general subjects were defined using newspapers’ section names.* |

The comparison of the versions shows high levels of agreement for most categories. The weighted average of Krippendorf’s *α* was .79 (see the detailed inter-coder agreement scores per category in the Supplementary Materials).

For the second reliability test we analyzed a corpus of 1.8 million news articles from *The New York Times* published between January 1995 and July 2017 to compare the results of the two versions at the article level (we expect this to be the more common use case).[[3]](#footnote-4) To do so, we aggregated the classifications of sentences into categories made by the classifier and applied them to the article level. Categories were assigned a percentage at the article level based on the number of sentences in which it appeared (see Online Appendix 3 for the details about this aggregation process).

We compared the classifications made by the two versions at the article level in two ways. We first measured the Krippendorf’s *α* and found that the alpha’s weighted average of was again high (*α*=.76). We also measured the correlation between the results using Pearson’s *r*, which showed a strong correlation (weighted average *r*=.81).

# Validation

Supervised methods offer a direct method for evaluating model performance by comparing the results of the classification method with a test set reserved before the training phase. We therefore evaluated our model using this test set, which resulted in accuracy measures for every category and on average. Then, as our solution is weakly supervised, we added more validations that are more common in unsupervised learning.

## Direct Assessment of Model Performances

We began the validation process using the held-out test set of about 6 million labeled sentences, in which most (95.1%) were originally labeled with a single expected category during our training-set compilation phase. After classifying the test set with our trained classifier, we identified multiple categories per sentence in most cases (80.2%), although this number was usually small (*M*=2.45, *STD*=1.13). To evaluate performance, we counted every classification as a true positive if one of the identified categories was true according to the test set.

The model achieved satisfactory levels of accuracy. Given the nature of the test set, and the fact that new topics were added without updating previous existing examples in the training set, we only have information regarding one expected label for each sentence in the test set in most cases. We do not know, for example, if the sentence is also relevant to categories that were added to the dataset later (as they did not exist at the time the sentence was added to the test set). We therefore do not have information regarding false positives (the model falsely identifying a category when it should not have). We have information only regarding false negatives (the model failing to identify a category when it should have) and true positives (the model succeeding in identifying an expected category).

Following this step, we calculated recall scores per category (the number of true positives divided by the sum of true positives and false negatives). As we did not have information about false positives, we did not calculate the equivalent precision scores (the number of true positives divided by the sum of true positives and false positives) (see the Supplementary Materials). We also have the overall number of true positive cases (where at least one of the identified categories was the expected one) and the overall number of assumed false positives (where none of the identified categories was correct, so we assume the sentence was falsely identified). We therefore calculated the average precision (*Precisionmean*=75.6%) and the weighted averaged recall (*Recallmean*=75.7%).

These results are consistent with accepted levels of accuracy despite the high resolution of the unit of analysis (i.e., sentence) and the large number of identified categories (*N*=651) (Grimmer and Stewart 2013). This finding suggests that our model provides a valid method for conducting a weakly supervised analysis of a large number of categories.

## Semantic Validation

Usually, weakly supervised learning models are evaluated by comparing their results with a benchmark dataset containing similar categories. As the discipline currently does not possess such a dataset, and our label definitions may differ from those of other researchers, we followed some of the validation steps used when validating topic models (e.g., Barberá et al. 2018). We provide two datasets that may reassure researchers that the model’s assumptions and predictions match its theoretical premises. First, we provide the test set used for the additivity test (section 5.2).[[4]](#footnote-5) Each row in the test set contains the tokens of the title of the article and the sentence analyzed in the training-set compilation phase, the expected label attached to the sentence during this phase, and the labels predicted by the two versions of the model.

Second, we provide a sample dataset of news articles analyzed during the additivity test. To create this dataset, we collected a sample of articles (up to 100) where more than 10% of the article was applicable to the category (this dataset contains only article titles, publication dates, and LexisNexis identifiers to allow for replication without violating copyrights). Although this is a relatively low threshold (in some cases only two or three sentences were classified into the category), it is usually sufficient to get a sense of the article’s main topics, which can then be validated using its title. In addition, to enable a more in-depth examination of these results, we provide similar results for this dataset at the sentence level to show the exact classifications made by our model.

## Predictive Validity

The process of assessing the model’s predictive validity is less time consuming than the process of semantic validation, and test whether the model is well correlated with external events for selected categories. Such an approach to validity (Quinn, Monroe, Colaresi, Crespin, and Radev 2010) requires a relatively consensus-driven and clear timeline of events to compare with in order to measure the precision and the recall of the model, i.e., to ensure the predicted spikes in the category's timeline are related to relevant events and that the model did not miss any major events. Here, we illustrate the predictive validity for four categories.

To perform this test, we analyzed a corpus of news articles from *The New York Times*. We aggregated the resulting classifications by averaging their scores from the article to the daily level, and from the daily to the monthly level. This resulted in a measurement for monthly media attention per category.

Figure 3 shows two categories representing specific events with a relatively easy-to-define timeline. The upper chart shows the presidential primary elections in the United States, which occur every four years. As expected, we can see a repeating lower attention / higher attention sequence: when an incumbent president is running for office, his or her victory in the primary election is almost certain; therefore, it attracts less attention.

The lower chart shows a category representing scandals and investigations related to President Bill Clinton. The main spike clearly indicates the Lewinsky scandal, with additional smaller spikes during the 2000 and the 2016 elections (when Senator Hillary Clinton ran for office). The small spikes before 1998 called for a closer examination. We filtered all pre-1998 news articles that our method labeled with this category and reviewed their titles. This analysis showed that these news articles did in fact deal with various investigations relevant to President Clinton (see the full list in the Supplementary Materials).

Another type of predictive validity is illustrated in Figure 4. The figure shows the monthly media attention paid to two seasonal categories, for which we can expect to find an annually repeating pattern. To show this cycle, we collapsed the 23 years of data into one calendar year, in which each data point represents a single year-month of data. We used sports categories as exemplars of expected periodical patterns—the categories of United States winter sports and American football—under the assumption that these categories will correspond to the seasonal calendar. The upper chart shows the category of winter sports, which are much higher during the winter months in the United States than the rest of the year. The lower chart shows the category of American football. This category also follows the expected periodic cycle, representing the beginning of the season in September and its end with the Super Bowl in late January or early February.

|  |
| --- |
| Figure 3 Predictive Validity Over Time |
| nyt_2_charts2016 ElectionsLewinsky scandal2016 Elections2012 Elections2008 Elections2004 Elections1996 Elections2000 Elections2000 ElectionsVarious investigations |
| *Note: The Y axes represent media attention to a category per month.*  |

# Conclusion

|  |
| --- |
| Figure 4 Predictive Validity by Seasonality |
| nyt_seasonal_scatterSeason beginsSeason beginsSeason endsSeason ends |
| *Note: The Y axes represent media attention to a category per month. The X axes represent the same month in every year.* |

Labeled datasets are the most important component for advancing the process of automatic meaning making. However, researchers struggle to gain access to labeled texts. In this paper, we offer a very effective and efficient method for generating labeled texts and show how researchers can use it for large-scale text analyses. The method proposed in this paper benefits from advances made in topic modeling to develop a low-cost method of topic analysis that meets the needs of theory-driven research: a collaborative, reusable, and additive method.

Throughout the training process, we used three types of topic analysis methods, each of which defines topics slightly differently. We first utilized topic models, which define topics as distributions over the vocabulary. We then converted the outputs of the topic models to topic association scores and created clusters of sentences, each representing a category. Last, we labeled these clusters and aggregated them into a training set, which we used to train a weakly supervised classifier that calculates the weights of features to predict each category based on the entire training set.

We do not claim that a topic originally identified by the topic model is identical to its corresponding cluster of sentences; the results of the topic model might differ from the results of the supervised classifier. However, we find that this process is well suited to our goals. Specifically, the combination of unsupervised and supervised methods allowed us to inductively and efficiently learn how categories are represented in the news, to add more categories or to further divide categories, without the need to retrain and relabel a new topic model. Our method enables researchers to first explore a corpus inductively using a topic model and then embed their topic model in a deductive topic analysis method with a larger list of supported categories. We believe this should allow for collaboration between different research projects and that it will help researchers to test more complex theories by incorporating increasing numbers of categories and variables into their theoretical models.

Combining unsupervised and supervised methods entails some caveats. For example, the supervised method might create the illusion that validation through comparison with a test set would be relatively simple. However, this test set was automatically created, and therefore should be treated with care, and the method should be further validated through other means. In addition, preprocessing choices should be appropriate to the method, and may differ between the two phases of our proposed solution. For example, when training a topic model, it is very common to remove stop words, while training a deep learning classifier does not necessarily include this step. Therefore, removing stop words before training the topic models may lead to a failure to identify a potential difference between related topics (such as two perspectives on the same topic, or two different styles, such as discussing the same topic with different levels of confidence, or from a personal or a collective perspective). In our illustration, we implemented the common preprocessing method used for topic models in order to show how common methods for training such models could be used. Nevertheless, we believe these aspects worth further investigation and experimentation in future research.

The suggested method is composed of multiple steps, some of which require specific choices of algorithms and configurations. Ours is not the only possible combination, and other clustering methods may be used in place of the one we developed. Our focus in this paper was not on creating a better topic model or even a context-aware clustering method, but rather on showing how such a combination of methods might be used to create a low-cost and additive method for a large-scale topic analysis with a high degree of resolution and a large number of categories. However, we do believe that the use of LDA as the starting point for our solution makes it much more relevant and accessible to researchers.

Compared with our pilot study, in which we manually labeled sentences, the advantages of our proposed approach are very clear. We were able to label over 30 times more categories and 5,000 times more sentences with the same amount of human labor. We achieved this goal by leveraging context both in the compilation of the training set and in the weakly supervised classifier architecture (i.e., by incorporating the title). In addition, the low cost of compilation allowed us to create a very large dataset of labeled sentences, which make it possible for us to use deep learning as the classification method. Last, our use of multilabel classification at the sentence level also contributed to more accurate and realistic sentence classification. Given the demonstrated capability of the model to incorporate additional topics and to refine the training set, we believe this approach could be of great use to the discipline.

1. In theory, other topic models could be used. We chose LDA as it is currently popular and requires fewer theory-specific assumptions (such as the involvement of a covariate variable in STM). [↑](#footnote-ref-2)
2. See nlp.stanford.edu/projects/glove/. [↑](#footnote-ref-3)
3. We will provide the analyzed dataset and code upon publication. [↑](#footnote-ref-4)
4. Upon publication. [↑](#footnote-ref-5)