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

**Abstract**

The collaborative effort of a 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. Additivity also enables the reuse of previous efforts or the merging of separate research projects, thereby increasing the accessibility of such methods and the ability of the discipline to create shareable content analysis capabilities. This paper proposes a weakly supervised topic analysis method, which combines 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 is a solid starting point for a low-cost and additive solution for a 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 increasingly find research methods focused 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 continues two endeavors recently made in this field, both aiming towards an accessible textual analysis method that can advance empirical research. The first is the reduction in costs, introduced by using topic models, as an unsupervised topic analysis method that eliminates the need to manually code large amounts of text (Blei 2012; Quinn, Monroe, Colaresi, Crespin, and Radev 2010). The second is the ability to explore and test theories, for example, by measuring the relations between topics and external 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 in these two vectors. First, it allows for more-accessible research, as researchers can collaborate on projects and identify topics from different domains while reusing existing trained models. Second, it expands the ability to test theoretical relations between variables, as it allows for the adding of more topical variables to the theoretical model (e.g., testing whether a relation between variables holds while controlling for other variables). Last, it increases the ability to analyze a different and possibly more general corpus, which strengthens the generalization power 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 single topic analysis method, thus resulting in an additive yet accessible topic analysis method.

The outline of this paper is as follows: section 1 reviews current methods and their limitations; section 2 describes the general idea behind our solution; section 3 presents the compiling of a training-set using unsupervised learning; section 4 describes the supervised classifier; section 5 demonstrates and validates the additivity of our solution; section 6 presents additional validations to the entire model; and section 7 concludes the paper, highlighting the advantages of our solution.

# Current Methods of Large-Scale Content Analyses

As a computational content analysis method, topic modeling allows for a large-scale analysis, which 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 cooccurrence 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 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, as topic models learn topics inductively instead of being given an a priory list of topics, they are sometimes difficult to use when testing a theory involving specific topical variables, 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, even a small variation in processing steps or the model’s configuration may result in different outcomes. Therefore, the reliability, stability and reproducibility of topic models raise a challenge. This challenge further increases when the corpus is not fixed but continuously grows, such as 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, possibly creating disagreements between researchers regarding the results of their analyses (Maier et al. 2018). As a result, the models’ contribution for the collaborative and replicable scientific efforts is compromised. Some of these limitations could be resolved if topic models could allow for the addition of topics to an existing model. Unfortunately, topic models do not offer a simple method for adding topics to an existing model (Blei 2012; Schwartz and Ungar 2015).

A more appropriate method for the classification of known categories is a dictionary analysis, in which a set of terms is searched for in the text 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 highly costly, and adding categories to an existing dictionary may carry even higher costs, as every new category should consider all other categories to prevent contradictions (Quinn, Monroe, Colaresi, Crespin, and Radev 2010). In addition, the accuracy of a 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 choice of words, to improve recall, but these approaches increase the dictionary start-up cost even further (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 compared with a dictionary analysis (Cambria and White 2014; Grimmer and Stewart 2013). It also offers simple mechanisms to add categories by 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 with the ability to update the list of categories in a more reliable, stable and reproducible manner.

Despite its advantages, studies in the social sciences usually use supervised learning only to identify small numbers of categories because it incurs high startup costs per identified category (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 a theory-driven research, the high costs it incurs limits its use by social scientists, especially when the tested theory involves more than a few variables.

# Weak Supervision as an Additive Alternative

The solution we suggest in this paper belongs to the family of weakly supervised methods. These methods reduce manual labor by splitting the training process into two phases. They first utilize a low-cost labeling method on raw data to minimize human labor while creating a training-set with labels that are not complete or fully accurate, yet useful. Then they train a regular supervised or a semi-supervised learning method on this training-set to create a deductive predicting method (Hernández-González, Inza, and Lozano 2016; Zhou 2018). In this way, these methods can reduce the cost of human labor, thus leveraging a very large training-set, performing 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 science (Deriu et al. 2017).

In our solution, we used unsupervised learning on a large volume of news articles to compile a training-set, and then used it to train a separate supervised classifier. We do note, however, that other low-cost methods can be used as alternatives for human coding, such as crowed-sourcing (Dehghani, Zamani, Severyn, Kamps, et al. 2017; Rudkowsky et al. 2018). One core contribution of our method is a better use of the available resources, so if we have more funding, we can use crowd-sourcing to create a training-set, and if we have less funding but more access to experts we can ask them to verify and interpret the outcomes of an 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, in a pilot study we performed with a group of six undergraduate coders, approximately three months of manual labor were required to compile a dataset of 10 thousand labeled sentences with reasonable inter-coder reliability for less than 20 categories. This amount of coding labor made the crowd-source alternative less attractive for our case, in which the number of categories to be coded was expected to be much higher.

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, thus increasing the accessibility of our solution. The main innovation of our solution is the way we converted the outputs of the topic models to create a labeled training-set (described in section 3.3). This conversion enables the weakly supervised solution. By moving from topic modeling to supervised learning, we were able to add categories to the training-set and train a supervised classifier to identify existing and new topics, without the need to retrain and relabel the original topic models (see demonstration in section 5).

# Training-set Compilation

Our solution is composed of two main phases: compiling a training-set and training a supervised learning classifier. The training-set compilation phase consists of the following steps: (1) collect corpora of texts (news articles in our case), each with a 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 labeling the clusters; and (6) add the labeled sentences to the training-set (see Figure 1 for a schematic visualization of the entire process). In the following, we describe the process in detail, while illustrating it with a large-scale analysis of news articles.

## Collect Corpora of Articles, Each with a General Subject

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 decompose to specific categories (e.g., move from Politics to Elections, Policy and Political Campaigns). In addition, technically, we found it preferable to train a topic model on a collection of news articles relevant to a single general subject, as such a corpus makes it easier to interpret 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 in the Supplementary Materials). For each general subject we looked for all sufficiently similar newspapers’ section names (e.g., economy, markets and finance), based on a collaborative judgment of three experts. We then collected all 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: General scheme of the process (section numbers in parentheses).* |

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

Before training each topic model, we performed standard preprocessing on each corpus separately: 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 between 25 to 100 topics), and 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)

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

A useful advantage of mixed-membership topic models, such as LDA or STM, is their fit for analyzing news articles, as those articles are more likely to contain multiple topics compares with other texts. However, this attribute also raises a challenge, as labeling and validating such topic models by reading entire articles is difficult when a researcher cannot exactly point to the part 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 examined the distribution of topics at the article level 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 obtain conclusive insights about the actual focus of the article.

Compared to articles, sentences tend to be more focused and hence associated with fewer topics. This fact simplifies their manual labeling as well as their use in training a supervised algorithm (Leetaru and Schrodt 2013). However, two sentences with similar content might have different meanings, depending, for instance, on the context of the article. We thus began the analysis at the article level and then moved to the sentence level before we labeled topics. This allows using the rich information at the article level while training the topic model, before moving to the sentence level.

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

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 corresponding *phi* of each word w in the sentence (*φk,w*):

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

The result is a 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 content (see the follow up example in section 3.5).

## Create Clusters of 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 expensive task. We achieved this goal by creating clusters, automatically created groups of sentences, that could be labeled collectively. To this end, we changed our perspective from sentences to clusters, where each cluster corresponded to a topic in the topic model. Therefore, instead of reviewing the various topic association scores assigned to a specific sentence, we reviewed 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 standardize topic association score above two (that is, the top 5% from all sentences), which we used as a minimal threshold for collecting sentences to a cluster. These groups were then reviewed by the human experts.

## Manually Labeling the Clusters

Human experts had 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 a contextual reading of sentences, we provided the experts with the entire article and title for each sentence. To ensure a coherent clustering of sentences, we asked the experts to set up the exact threshold for each cluster. This less typical role of human coding was done by arranging the collected sentences in five groups based on their standardize topic association score (ranging between 2–2.5, 2.5–3, 3–3.5, 3.5–4 and above 4). We then asked the experts to point at the exact threshold that would enable a coherent cluster, per topic. Moreover, the label and threshold needed to correspond to each other, as a broader and more general label might lead to choose a lower threshold that will include more sentences. For example, say sentences with the highest topic association score per a given topic are all related to US-Russia relations, yet lower association scores also include sentences related to US-Mexico relations. It is up to the human experts to decide whether to choose a higher threshold and label the topic “US-Russia relations”, or choose a lower threshold and a broader label, such as “US foreign affairs”. This process sometimes demanded 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 that ran away from a sheriff and caused a car accident. The result of the original topic model we trained was a distribution at the article level with two main topics. Our goal was to look for texts that are more strongly associated with each topic. When we moved 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 common 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 were no longer care for the original results of the topic models. For once, the topic association score was not used later in the process, which implies we used only binary tags (i.e., related to the category or not, though a sentence could have been tagged as related to more than one category). In addition, some words, such as stop words, were removed during preprocessing, therefore were not given a phi value by the topic model nor contributed to the topic association score of their sentence. However, the clustering and manual labels inference were preformed using original sentences, including all words. Both these decisions were made to separate between the training-set compilation phase and the training of the supervised learning classifier.

## Add 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 the 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 to a single training-set. In cases where the labels given for topics learned by one topic model overlapped labels from another topic model, we merged both groups to a single group of sentence with one label.

The purpose of the entire process is to train a supervised classifier, therefore the 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, assisted in training human coders and fine-tune the process. We therefore recommend researchers who are interested in applying the suggested process to follow this evaluation, which we describe in more details in Online Appendix 1.

# Designing and Training a Deep Learning Sentences 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 usually depend on large amounts of data, sometimes millions of labeled examples (LeCunn, Bengio, and Hinton 2015). This is probably one of the barriers for using deep learning in computational social sciences, especially where the goal is to identify large number of categories. Yet it is also where we 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 is a good demonstration of a sufficient method. To keep the discussion concise, we describe the classifier in short. (For detailed explanations see Online Appendix 2. We recommend researchers who are new to the field of deep learning to review the appendix before moving to the section).

## Preprocessing Sentences

Because deep models automatically learn the representation of raw data, the preprocessing of text input is somewhat different from classical machine learning. 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 discussed 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 this length to be of 100 words (including punctuation marks), which covered 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 practically ignores. The input layer of our model then embedded 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 the sequential information, or order of words, and 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 to allow it to store an entire sentence in memory. 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 reasonable 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 first two layers: 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 set to the number of categories plus 30, 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.

We believe that even though a sentence is usually more focused than an entire article, it still may refer to more than one category, especially when categories are not mutually exclusive. In fact, when it comes to the political domain it is rather common that one sentence will contain multiple topics (consider, for example, a political debate on public expenses on health).

We therefore designed the output layer of the model to predict a multilabel classification, where 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 giving the probability of each category to be true in separate. All categories with a probability of more than 50% to be true 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 each layer size (number of modules) were set, we tuned the hyperparameters. To reduce the risk of overfitting that may 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 stopped the training before performance degraded 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 may simply add manually labeled text examples. Alternatively, the researcher can add more iterations of our method: collect an additional corpus with a general subject, train a topic model, convert its outputs to clusters of sentences, infer a label to each cluster, and add the sentences with the new labels to the training-set.

## Illustrating Additivity

An interesting option to utilize additivity is to decompose one of the existing categories to even more focused 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 (named version 15) was trained to identify 450 categories. One of the categories was Guns and Gun Control in the US, which we extracted from corpora collected using media section names, such as Politics and Crime. For the sake of this demonstration, we wanted to decompose this category to higher resolution categories, such as the attitudes towards gun control and actual use of guns in the US. These nuanced-based categories allow further development of empirical research on this topic. To decompose it, we run the trained supervised classifier and used it to identify news articles that discussed the category of Guns and Gun Control in the US (e.g., all articles in which this category was identified with more than 10%). By doing so, we created a new corpus that was focused on gun control in the US.

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

To allow more variance of relevant topics, we have repeated this process and decomposed two additional categories, “Elections & Primary Campaigns,” and “Conflicts”, which assumed to be relevant for our interest in gun control. We have also collected another corpus from the Opinions section in various newspapers, to add more perspectives on potential 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, and ended up with a training-set containing about 100 million sentences, labeled with 651 topics in total (see Table 3). We used this training-set to train a new supervised classifier (version 16).

The entire training-set compilation phase, including the addition of these new 201 categories (from a training-set of 100 million sentences), required approximately 400 work hours, performed by human experts who devoted their efforts to infer a label and set a threshold to each cluster. As we have seen in our pilot study mentioned above, 400 work hours have resulted in a dataset of 10 thousand labeled sentences for less than 20 categories only.

## 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 held-out test-set of 5.99 million sentences, sampled from the compiled training-set. As we do not have a gold standard of human coding that would have allowed an external comparison of accuracy, we treated the two versions as two coders and tested their inter-coder reliability. We did not expect to find full agreement as any addition of categories to the model might affect the model classification of other related categories. Our expectation was to find high enough agreement between the two versions to indicate the stability of the model given 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 |
| Opinions | 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 |
| Tech | 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 shows high levels of agreement in most categories. The weighted average of *Krippendorf’s α* was .79 (see 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 two versions at the article level (the usage we expect to be more common).[[3]](#footnote-4) To do so, we aggregated the identifications made at the sentence level to the article level. Based on the number of sentences in which a category has appeared, it was assigned a percentages at the article level (See Online Appendix 3 for the details of this aggregation).

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). Then, we measured the correlation between the raw results using *Pearson’s r*, which also showed a strong correlation (weighted average *r*=.81).

# Validation

Supervised methods offer a direct evaluation of model performances by comparing the results of the classification method with a test-set put aside before training. We first present accuracy measures for every category and on average. Then, as our solution is weakly supervised, we add more validations that are more common in unsupervised learning.

## Direct Assessment of Model Performances

We started the validation using the held-out test-set of about six 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%), but this number was usually small (*M*=2.45, *STD*=1.13). To evaluate the classifier performances, we counted every classification as a true positive if one of the identified categories was true according to the test-set.

The model reached 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 usually have information only regarding one expected label for each sentence in the test-set. We do not know, for example, if that sentence is also relevant for 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 (when the model falsely identified a category when it should not have). We have information only regarding false negatives (when the model failed to identify a category when it should have) and true positives (when the model succeeded in identifying an expected category).

Following this step, we calculated recall scores per category (number of true positives divided by the sum of true positives and false negatives) but not precision (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 our model is at least a firm starting point for a weakly supervised analysis of a large number of categories.

## Semantic Validation

Usually, evaluating weakly supervised learning models is done by comparing their results with a benchmark dataset containing similar categories. As the discipline currently does not possess such a dataset, and our definition for various labels may differ from 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 the model’s assumptions and predictions match its theoretical premises. First, we provide the test-set used in the additivity test (section 5.2).[[4]](#footnote-5) Each row in the test-set contains the tokens of the title and sentence analyzed in the training-set compilation phase, the expected label attached to it during this phase, and the labels predicted by the two versions of the model.

Second, we provide a sample dataset of news articles analyzed in the additivity test. To create this dataset, for each category, we collected a sample of articles (up to 100) at which the category crossed a threshold of 10% (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 identified using the category), it is usually enough to get a sense of the article's main topics, which can then be validated with its title. In addition, to enable a more in-depth examination of these results, we provide similar results on this dataset at the sentence level to show the exact classifications made by our model.

## Predictive Validity

Another, less time-consuming, process is assessing the predictive validity of the model. It shows to be well correlated with external events on selected categories. Such a validity approach (Quinn, Monroe, Colaresi, Crespin, and Radev 2010) demands a relatively agreeable and clear timeline of events to compare with, to measure both 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 missed no major event. We illustrate here the predictive validity of four categories.

To perform this test, we analyzed the corpus of news articles from *The New York Times*. We aggregated the resulting classifications to the daily level, then to the monthly level, by averaging them. The result was a measure for the monthly media attention per category.

Figure 3 shows two categories, which represent specific events with a relatively easy-to-define timeline. The upper chart shows the US Presidential primary elections, which occur every four years. As expected, there is a repeated pattern of a lower attention-higher attention sequence: when an incumbent President is running for office, his victory in the primary elections is almost certain; therefore, it attracts less attention.

The lower chart shows the category of scandals and investigations related to President Bill Clinton. The main spike refers clearly to the Lewinsky scandal, also echoed in 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 news articles in which our method identified this category prior to 1998 and reviewed their titles. The result shows these news articles deal with various investigations relevant to President Clinton (see the full list in the Supplementary Materials).

Another type of predictive validity is demonstrated in Figure 4. The figure shows the monthly media attention paid to two seasonal categories, where we can expect to find the same pattern every year. For this purpose, we collapsed the 23 years of data into one calendar year, in which each data point represents a single year-month data. We have used our sport categories as an exemplar of expected periodical patterns. We show the categories of US Winter Sports and American Football, under the assumption that these categories will be correlated with the yearly seasonal calendar. The upper chart shows the category of winter sports, where the US winter months are much higher than the rest of the year. The lower chart shows the category of American football. This category also follows the expected periodical 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 by Time Line |
| nyt_2_charts2016 ElectionsLewinsky scandal2016 Elections2012 Elections2008 Elections2004 Elections1996 Elections2000 Elections2000 ElectionsVarious investigations |
| *Note: The Y axes represent the media attention of a category per month.*  |

# Conclusion

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

Labeled datasets are the basic element that can promote automatic meaning making. However, we are always short of labeled texts, asking for more than we have. In this paper, we offer a very effective and efficient source for 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 models to develop a low-cost method of topic analysis that fits the characteristics of theory-driven research: a collaborative, reusable and additive method.

Throughout the training process, we used three types of topic analysis methods, each define 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 represents a category. Last, we labeled these clusters and aggregate those to a training-set, then used it to train a weakly supervised classifier, which calculates 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 accomplishes our goals very well. Specifically, the combination of unsupervised and supervised methods allowed us to inductively and efficiently learn how categories are represented in the news, add more categories, or further decompose categories, without the need to retrain and relabel a new topic model. For example, our method enables researchers to first explore a corpus inductively using a topic model and then embed their topic model in a larger, deductive topic analysis method. We believe this should allow for a collaboration between different research projects, and contributes to researchers’ ability to test theories that are more complex, by incorporating increasing numbers of categories and variables to their theoretical models.

Combining unsupervised and supervised methods carries some cavities. For example, the supervised method might create the illusion that validation would be simple through a comparison with a test-set. Yet this test-set was automatically created, therefore should be treated with care, and the method should be validated through other means as well. In addition, preprocessing choices should fit the method they serve, and might differ between the two phases of the suggested solution. For example, when training a topic model it is very common to remove stop words, yet training a deep learning classifier does not necessarily include these steps. By performing these steps before training the topic models, we might have missed a potential separation between related topics (think of two perspective of same topic, or 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 followed the common preprocessing used for topic models, to show how common training of such models could be used in our method. Nevertheless, we believe these aspects worth further investigation and experimentation in future research.

The suggested method is composed of multiple steps, some with specific choices of algorithms and configurations. This is not the only possible combination, and other clustering methods may replace the one we developed. However, our focus in this paper was not on creating a better topic model or even a context-aware clustering method. Our main aim was to show 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 resolution and a large number of categories. Choosing LDA as the starting point of this solution makes it much more relevant and accessible to a community of researchers. Furthermore, compared with our pilot study in which we manually labeled sentences, the advantages of the suggested approach were 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 the 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 compilation allowed us to create a very large dataset of labeled sentences, which enabled the use of deep learning as the classification method. Last, the multilabel classification at the sentence level also contributed to a more accurate and realistic classification of sentences. Given the demonstrated capability of the model to incorporate additional topics and refine the training-set, we believe this path could be of great use to the discipline.

1. In theory other topic models could be used. We chose LDA as it is currently more popular and carries less theory-specific assumptions (like the involvement of a covariate variable in STM). [↑](#footnote-ref-2)
2. 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)