Latent Measurement from Text Data

Principles and Practices for Unsupervised and Weakly Supervised Approaches (WORKING DRAFT)

Quantitative and Computational Methods for the Social Sciences

DOI: 10.xxxx/xxxxxxxx (do not change)
First published online: MMM dd YYYY (do not change)

Colin R. Case
University of Iowa

Rachel Porter
University of Notre Dame

Abstract: This Element develops a framework for constructing and validating latent measures from text using low-supervision scaling approaches. We organize the text-to-measure pipeline around three cross-cutting design choices—text inclusion, numerical representation, and scaling procedure—and demonstrate, through a systematic comparison of over one hundred pipeline configurations, that these choices interact to shape downstream measures in consequential ways. Using issue-specific positioning in congressional campaign platforms as an illustrative application, we show that credible measurement requires principled design at every pipeline stage paired with multifaceted validation.

Keywords: measurement; text analysis; weakly supervised learning; unsupervised learning; congressional elections

JEL classifications:

ISBNs: xxxxxxxxxxxxx(PB) xxxxxxxxxxxxx(OC)
ISSNs: xxxx-xxxx (online) xxxx-xxxx (print)

Contents

1 Introduction

Measurement lies at the center of empirical social science. Researchers routinely seek to represent latent concepts—such as beliefs, identities, and preferences—in ways that allow for comparisons across individuals, groups, and time. As digital text has become an increasingly important source of social data, researchers have turned to computational text analysis to build continuous, quantitative measures of these latent concepts. This shift toward computational approaches has broadened the range of measurable phenomena, the scope of analysis, and the contexts of empirical study.

Computational approaches to text-based measurement infer latent constructs indirectly by extracting construct-relevant signals from patterns in natural language. These signals are used to place units of interest—such as documents, speakers, or parties—along one or more latent dimensions, a process we refer to as *scaling*. The scaled positions of units are determined through a sequence of measurement choices that specify which features of language constitute construct-relevant signals. Each of these choices reflects assumptions about the relationship between language and the target construct being measured. Credible measurement, therefore, requires (i) a clear conceptual definition of a latent construct of interest, (ii) an explicit set of principles guiding measure construction, and (iii) a validation strategy for assessing whether scaled positions recover that target latent construct.

Computational text analysis encompasses a growing class of *unsupervised* and *weakly supervised* approaches for recovering latent dimensions from text and scaling units of interest along those dimensions. Unsupervised approaches recover latent dimensions entirely from statistical regularities in text. Weakly supervised approaches allow researchers to steer recovery toward a target dimension of interest.[1] We refer to these methods collectively as *low-supervision scaling approaches*, emphasizing that latent dimensions and scaled positions are recovered primarily from the text itself.

Low-supervision approaches contrast with annotation-driven *supervised* learning, where researchers begin with a target construct, use human coding to map text to categories of interest, and train predictive models to reproduce those judgments at scale on unlabeled text. Measurement quality in this paradigm hinges on producing large numbers of high-quality text annotations; however, this annotation process is typically time and resource-intensive, requiring human coder training and ongoing monitoring of reliability to prevent drift and systematic error.[2]

[1] "Unsupervised" refers specifically to the absence of researcher input in guiding latent dimension recovery; in practice, however, these methods still require researcher input to render scaled measures interpretable, as we discuss in Section 2.

[2] Although online labor markets such as Amazon Mechanical Turk reduce these burdens by enabling low-cost annotation at scale (Carlson & Montgomery, 2017), these platforms have grown vulnerable to quality degradation, fraudulent participation, and synthetic responses (Kennedy et al., 2020; Westwood, 2025).

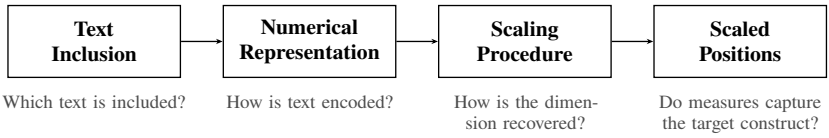


Figure 1 Overview of the text-to-measure pipeline

The relatively limited need for manual text annotation in low-supervision scaling, combined with declining technical barriers to implementation, has made these approaches increasingly attractive to researchers. Recent work uses low-supervision approaches to measure a variety of continuous latent constructs, including ideology (Case, 2027; Napolio, 2025), party unity (Motolinia, 2025), narratives (Kitagawa & Shen-Bayh, 2024), emotions (Widmann & Wich, 2023), reputations (Bellodi, 2023), threat (Wong, 2026), moral foundations (Duan et al., 2025), and conflict (Finke & Risse, 2024). Yet the growing use of low-supervision scaling has outpaced the development of shared standards for measurement construction and validation. Measurement design for low-supervision scaling can take many forms, but there is little systematic guidance on which designs are likely to perform well for a given task.

To construct credible measures of latent constructs from text requires navigating a series of interconnected decisions throughout the *text-to-measure pipeline*: the sequence of stages that transform raw text into quantitative measures. Existing research has explored how these decisions affect measurement either by assessing the cumulative impact of choices across pipeline stages (Grimmer, Roberts, & Stewart, 2022; Grimmer & Stewart, 2013; Park & Montgomery, 2025; Ying, Montgomery, & Stewart, 2022) or by examining the consequences of particular design decisions at individual stages of the pipeline (Denny & Spirling, 2018; Rodriguez & Spirling, 2022). We build on this literature by outlining a common framework for measurement-pipeline design tailored to low-supervision scaling. The framework serves two purposes: to clarify how choices at individual pipeline stages accumulate and interact to shape downstream measures, and to provide a principled basis for assessing their validity.

In this Element, we focus on three cross-cutting choices that arise in virtually any low-supervision scaling pipeline: (i) which texts enter the measure-construction process, (ii) how information in those texts is encoded in numerical form, and (iii) how that numerical information is used to locate units along a latent dimension intended to capture the researcher's target construct. We argue throughout the Element that these design choices—summarized in Figure 1—should be guided by how the target construct is manifested in the text under study. Because target constructs and their textual expression vary across research settings, we do not propose universal best practices. Instead, we aim to equip researchers to make principled design decisions and to assess systematically whether their resulting measures recover the intended latent construct.

To guide the reader through the process of developing a low-supervision

scaling pipeline, the Element is organized into four parts. Section 2 provides an overview of the text-to-measurement pipeline, emphasizing the key choices involved in defining documents, representing text, and determining the degree of researcher input that enters the scaling procedure. Section 3 turns to validation, developing an approach for assessing whether a given pipeline configuration recovers a researcher's intended latent dimension. To do so, we systematically compare more than one hundred pipeline configurations to identify where design choices most significantly shape measurement. We then use the regularities that emerge from these comparisons to derive guiding principles for pipeline construction, helping researchers narrow the set of potential pipeline configurations they consider for their own measurement tasks. Section 4 examines large language models within a measurement-pipeline framework, demonstrating that these models do not eliminate the core challenges of pipeline design and that validation remains crucial. The Element concludes by outlining the responsibilities researchers assume when constructing latent measures with low-supervision approaches.

1.1 Motivating Example: Measuring Issue Positioning

Producing estimates for political actors' latent issue positions—what some label "ideologies" or what we will refer to as political orientations—is a core methodological interest in the social sciences.[3] These measures are conventionally understood to capture where candidates, legislators, parties, and voters fall on a left-right political spectrum and are essential for evaluating theories of political behavior, party competition, and democratic responsiveness. A substantial literature has developed in response, producing estimates for various actors' political orientations using roll-call votes, campaign contributions, survey responses, speeches, manifestos, and social media networks (e.g., Ansolabehere, Snyder, and Stewart 2001; Barbera 2015; Bonica 2014; Laver, Benoit, and Garry 2003; Poole and Rosenthal 1985; Slapin and Proksch 2008).

Although these measures derive from different data sources, they share a common structure: each locates an actor at a single point on a left-right scale. Put differently, they compress an actor's full range of issue positions into a single measure. Such measures are straightforward to interpret and align closely with spatial models used in theoretical work on elections, legislatures, and public opinion.[4] But this representational convenience can also come at a cost. Collapsing actors' political orientations into a single left-right summary measure may obscure meaningful variation in how they position themselves across issues. When actors exhibit positional constraint—such that their left-right stances

[3] We do not assume politicians' expressed issue positions reflect underlying policy preferences. We therefore avoid using the term "ideology," which is generally understood to imply sincere policy preferences. Our "political orientation" terminology follows Tausanovitch and Warshaw (2017).

[4] Spatial models of legislative behavior, electoral competition, and public opinion formation typically assume that actors can be located in a low-dimensional policy space, often a single left-right dimension (e.g., Aldrich and McKelvey 1977; Downs 1957; Poole and Rosenthal 1985).

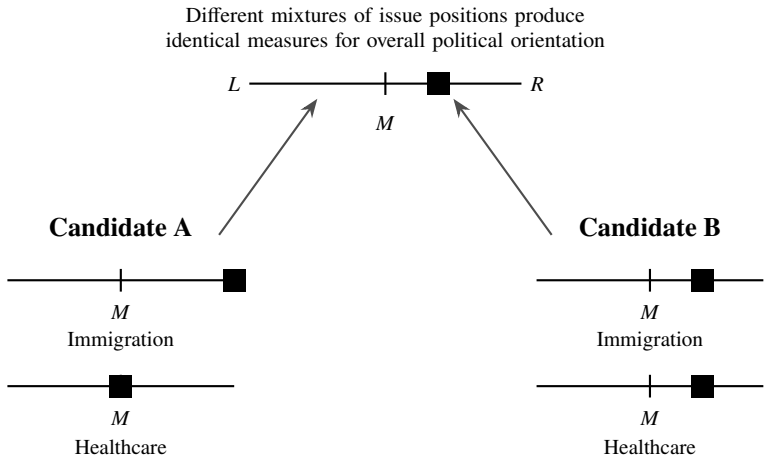


Figure 2 Single-dimensional measures for political orientation

are similar across issues—measures of overall political orientation provide an efficient way to summarize where actors stand across issues. Conversely, when actors exhibit positional heterogeneity—such that their left-right stances vary across issues—such measures instead obscure how positions are configured across issue domains (Broockman, 2016).

Figure 2 illustrates this measurement limitation. The top horizontal line represents a summary left-right axis for overall political orientation, where *M* denotes the midpoint and *L* and *R* denote the leftmost and rightmost political orientations, respectively. The lower scales show issue-specific positions for two candidates along left-right axes. Candidate A adopts a moderate position on healthcare (near the midpoint) but an extreme position on immigration (far to the right), whereas Candidate B takes consistently right-leaning positions on both issues. Because Candidate A's moderate and extreme positions offset one another when compressed into a single score, both candidates are assigned a similar overall political orientation despite meaningful differences in their issue-specific positioning, rendering them largely indistinguishable.

This pattern is not merely hypothetical. At the party level, European populist right parties such as Italy's Five Star Movement often combine centrist or left-leaning positions on welfare with far-right stances on immigration and cultural issues. Similar heterogeneity is evident at the level of individual politicians. During her first congressional campaign for the U.S. House of Representatives in 2014, Elise Stefanik (R-NY) took left-leaning positions on some issues—advocating for clean energy and increased federal funding for higher education—while taking far-right stances on others, such as opposing any restrictions on gun ownership. Summary measures of political orientation obscure these forms of issue-specific positional heterogeneity.

Understanding where political actors stand across various issue domains matters for core questions of democratic politics. Do legislators represent their constituents' policy-specific preferences? How do parties differentiate themselves in electoral fields? Do voters select candidates based on policy proximity? Which issues drive cross-party coalition formation? Is polarization uniform across issue domains or concentrated in specific areas? Answering such questions requires measures that capture how positions are configured across issue domains—among legislators, parties, and voters alike.

For the purposes of this Element, we focus on estimating left-right, issue-specific positions for a specific group of political actors: candidates for the U.S. House of Representatives. These measures can serve as inputs for empirical analyses addressing longstanding questions about candidate strategy, voter information, and political representation. In particular, they can inform assessments of constituency representation (Case & Porter, 2026; Fenno, 1978), differences between elites and the mass public in positional coherence (Converse, 1964; Marble & Tyler, 2022), voters' ability to infer candidates' policy stances from partisan cues (American Political Science Association, Committee on Political Parties, 1950; Hetherington, 2001), and the connection between candidates' positions and voter decision-making (Colao, Broockman, Huber, & Kalla, 2025; Downs, 1957). Despite their relevance for such questions, these measures remain largely unavailable for congressional candidates, especially in forms that capture issue-specific positioning at scale.

The measurement task, then, is to construct continuous, issue-level measures of candidate position-taking in campaigns. To demonstrate the cross-cutting choices involved in text-based measurement with low-supervision scaling, we use this task as a running illustration throughout this Element. Our data consist of campaign platforms drawn from candidate websites for U.S. House races, which provide direct statements of candidates' issue positions in their own words. This application is well-suited to our purposes for several reasons:

1. **Flexible document definition**: Campaign platforms can be analyzed at multiple levels—full documents, platform points, or individual paragraphs—illustrating how text inclusion shapes downstream measures.

2. **Linguistic heterogeneity**: Candidates articulate positions through varied strategies across issue domains. This variation provides a useful testbed for comparing how scaling approaches perform across different linguistic environments within the same measurement task.

3. **Multiple validation benchmarks**: The polarized U.S. political context provides clear expectations about left-right cleavages within issue domains and offers multiple complementary benchmarks for validation.

4. **Task complexity**: Inferring left-right, issue-specific positions from political text is a conceptually demanding measurement task, creating strong incentives for lower-supervision scaling approaches.

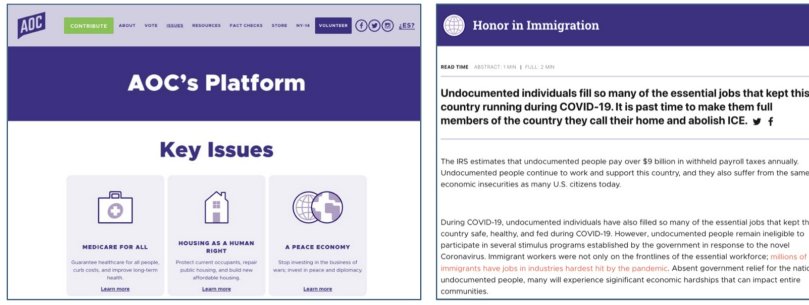


Figure 3 Example Campaign Website: Alexandria Ocasio-Cortez (D-NY)
Source: https://www.ocasiocortez.com/issues

The sections that follow develop each of these points in greater detail, providing the conceptual grounding and practical guidance necessary to understand and evaluate the illustrative example as it recurs throughout the Element.

1.1.1 Data: congressional campaign platforms

We draw on CampaignView, a database of U.S. House candidates' campaign platforms collected from campaign websites, which is well-suited to measuring issue-specific candidate positioning (Porter, Case, & Treul, 2025). Campaign platforms are the policy agendas candidates present to voters during elections. Existing research demonstrates that these platforms exhibit meaningful variation in issue positioning (e.g., Bailey and Reese 2025; Case 2027; Case and Porter 2026; Druckman, Kifer, and Parkin 2009; Porter, Treul, and McDonald 2024), indicating that candidates actively tailor their position-taking rather than simply adopting party positions. The CampaignView corpus covers 5,897 (76.2%) of the 7,739 major-party, ballot-eligible candidates who ran in primary elections for the U.S. House of Representatives between 2018 and 2024, providing broad coverage across parties, regions, and election cycles.

Figure 3 depicts the structure of a typical campaign website using the site of Alexandria Ocasio-Cortez (D-NY) for the 2022 election cycle. Congressional campaign websites organize platforms as a series of discrete platform points. The left panel displays Ocasio-Cortez's main platform page, where visitors select from a menu of issue domains, including *Medicare-For-All*, *Housing as a Human Right*, and *A Peace Economy*. The right panel shows the page that opens when a visitor selects one of these platform points, which contains a short heading identifying the issue domain and several paragraphs describing Ocasio-Cortez's position. Notably, the policy content within a single platform point often spans multiple topics: for instance, Ocasio-Cortez links immigration reform to social welfare in her discussion of *Honor in Immigration*. This reflects an authentic feature of campaign messaging, as candidates often weave together

discussions of distinct issue domains to serve their broader messaging goals.

The multi-topic nature of these texts raises a fundamental question pertinent to any low-supervision scaling task: which text should a researcher pass to their measurement pipeline in order to produce valid measures? In our measurement task, for example, a researcher might use either a candidate's full campaign platform or finer-grained text segments, such as individual platform points, to construct measures of issue-specific positioning. These text-inclusion decisions matter because they determine the mix of issue-relevant and non-relevant content supplied to the measurement pipeline. That mix, in turn, shapes the textual signal available for scaling and ultimately affects how well the resulting measures capture the issue-specific political positions as communicated by candidates. We return to these distinctions in Section 2.1, where we discuss text inclusion in greater detail, and in Section 2.3, where we describe how different scaling approaches can recover positional signal from text. Section 3 then shows how text inclusion interacts with downstream scaling to systematically shape how well the resulting measures reflect candidates' communicated issue positions.

Campaign platforms vary not only in how policy content is organized, but also in the linguistic forms through which candidates articulate their positions. Candidates adopt a range of position-taking strategies, including emphasizing problems or solutions, invoking policy details or abstract values, and critiquing opponents. This diversity of expression complicates text-based scaling because the textual signals associated with a common underlying left-right position may differ substantially across candidates.

Consider these excerpts from two Republican candidates discussing gun rights in campaign platforms:

> "The Constitution states the right of the people to keep and bear Arms, shall not be infringed...I believe in the **ability to open carry nationwide**."
> — Christy McLaughlin (R-FL), 2020 campaign platform

> "Gun Ownership: Al Lemmo believes there **should be no restrictions** for law-abiding citizens to own the firearms of their choice."
> — Al Lemmo (R-MI), 2020 campaign platform

Both candidates adopt strongly right-leaning positions on gun policy, but they emphasize different policies: McLaughlin stresses the right to carry firearms across jurisdictions, whereas Lemmo focuses on removing all restrictions on acquisition. Measuring positional meaning in issue-specific contexts, therefore, requires moving beyond the particular policies candidates invoke to the broader left-right orientation those policies convey.

A distinct but related challenge arises when candidates express substantively equivalent positions using different language. In Ocasio-Cortez's platforms, for example, she calls for *Medicare-for-All*, but similar positions could be expressed using alternative phrases such as single-payer healthcare or universal public coverage. These phrases advocate a common position—replacing private health

insurance with a federally administered system accessible to all Americans—but share little vocabulary. Approaches that fail to capture this semantic equivalence risk treating linguistic variation as a positional difference.

These forms of linguistic heterogeneity are not exhaustive of potential sources of measurement difficulty, but they illustrate the central challenges that many text-based scaling approaches must confront. In low-supervision scaling, measurement quality depends on whether a numerical representation of text preserves the linguistic signals needed to accurately map diverse articulations of political position onto the latent dimension of interest. Section 2.2 discusses which numerical representations of text are better suited to preserving different forms of linguistic signal. Section 3 then evaluates the relative advantages of these different forms of text representation by examining the measurement tasks for which they best capture candidates' issue positions.

1.1.2 Validation benchmarks

Assessing whether text-based scaling methods recover genuine left-right differences requires benchmarks against which to evaluate *construct validity*. We produce issue-specific, left-right positioning measures across four domains: abortion, guns, healthcare, and immigration. Table 1 outlines the dimensions of left-right conflict that structure these domains. In contemporary U.S. politics, these dimensions are well-defined and consistently invoked; they have become the dominant structure of partisan debate on these issues, organizing how candidates differentiate themselves and how voters perceive their choices (Layman & Carsey, 2002; Layman, Carsey, & Horowitz, 2006). Candidates may discuss other policies—on healthcare, for instance, a candidate might address maternal mortality rates or the opioid crisis—but such discussions reflect topical concerns rather than dimensions of left-right conflict.

Definitional Concept: Construct Validity

Construct validity refers to the degree to which a measure supports the interpretation that it captures the theoretical concept—or construct—it is intended to represent. Because latent constructs are not directly observable, they must be inferred from observable indicators, and validity is evaluated through accumulated empirical evidence rather than direct verification. For treatments of construct validity that emphasize research design and quantitative measurement, see King, Keohane, and Verba (2021), Adcock and Collier (2001), and Jackman (2008).

The clarity of these dimensions provides a direct basis for evaluating construct validity. Issue-specific scaled positions should place candidates who advocate extreme positions—such as total abortion bans, the abolition of firearms sales, deportation without exception, or the elimination of private health insurance—at

Table 1 Scaled Dimensions of Left-Right Conflict by Issue Domain

Issue Domain	Left	Right
Abortion	Unrestricted access, codify the right to abortion.	Completely banned and fully restricted, no exceptions.
Guns	Ban private ownership, mandatory buybacks.	Unrestricted access to all firearms in all jurisdictions.
Healthcare	Fully nationalized, single-payer with universal coverage	Fully privatized, free-market system, no government role.
Immigration	Maximally permissive, abolish deportation, full amnesty.	Maximally restrictive, mass deportation, zero tolerance.

the poles of the respective issue's left-right spectrum. Candidates who defend the political status quo or adopt mixed policy arrangements should be assigned to more centrist positions. Still other candidates who advocate incremental reforms—such as expanding background checks and waiting periods for firearms purchases or permitting abortion only in cases of rape, incest, or threat to the life of the mother—should be assigned positions closer to their party's mainstream. When low-supervision scaling assigns similar scores to substantively opposing positions, reverses their ordering, or compresses clearly distinct stances, these errors are readily identifiable as failures of measurement rather than artifacts of conceptual ambiguity.

No single validity test is dispositive. As we demonstrate in Sections 3 and 4, evaluating construct validity across multiple benchmarks is essential because each benchmark is sensitive to different forms of measurement failure. A measure may succeed on one benchmark while failing on another, so only their combination can provide a credible basis for assessing whether scaled positions recover the intended construct.

1.1.3 Task complexity

Although the dimensions of left-right conflict in the issue domains we examine are conceptually clear, inferring candidates' positions from campaign platform text remains a demanding measurement task. Our objective is to locate candidates on a continuous, issue-specific left-right dimension rather than simply classify them as left- or right-leaning. In a fully supervised framework, this would require human annotators to make fine-grained, often subjective distinctions among candidates who fall on the same side of the political spectrum. Such judgments can be difficult to make reliably when candidates differ less in direction than in the intensity, scope, or configuration of their issue positions. Making these determinations would also require substantial training and detailed codebooks to define policy-specific terminology that may not be widely legible

to nonexperts. For example, support for red flag laws and opposition to the Hyde Amendment convey clear positions, but these associations may not be obvious to annotators without relevant domain knowledge. The cost of building such a supervised framework grows rapidly with the number of issue domains, election cycles, and candidates under study.

These challenges are not specific to our measurement task but arise more broadly in settings where cross-unit differences are inherently difficult to judge or require domain-specific expertise. For example, studies of violent political rhetoric (Kim, 2023) and partisan incivility (Munger, 2021) often reveal substantial disagreement among annotators over how particular texts should be interpreted and located within the relevant conceptual space. This reflects the cognitive demands of tasks that require annotators to carry a detailed mental map of the relevant domain, specifying not only categorical distinctions but also where particular instances fall relative to one another. Having annotators make pairwise judgments of text can reduce this cognitive burden (Carlson & Montgomery, 2017), but it does not eliminate the need for substantive expertise.

Low-supervision approaches offer a path forward by reducing reliance on large volumes of manually annotated data.[5] Our measurement task is precisely the kind of setting in which low-supervision methods are attractive, because they can leverage substantive knowledge without requiring exhaustive annotation. At the same time, they are not foolproof. The sections that follow examine when and why low-supervision scaling approaches succeed or fail at distinguishing candidates' positions across target dimensions.

2 Text-to-Measure Pipeline

Every text-based measure is the product of a sequence of decisions that collectively determine how qualitative language is quantified. Although the specific design choices researchers employ at each stage of this text-to-measure pipeline vary widely, the pipeline itself has a common structure organized around three main decisions: defining the documents that enter the pipeline, encoding their content as numerical representations, and determining how those representations are used to position units along a target latent dimension.

This section describes this text-to-measure pipeline conceptually, explaining the logic and potential consequences of the cross-cutting choices researchers face at each pipeline stage. We use the measurement task introduced above—producing issue-specific, left-right measures for congressional candidates' position-taking from campaign platform text—as a running illustration, but the pipeline generalizes to any setting in which researchers seek to construct quantitative measures of latent concepts from text using low-supervision methods.

[5] Low-supervision methods avoid the costs and cognitive constraints of large-scale human annotation, but they do not resolve the underlying semantic ambiguity of political text. Because these methods rely on statistical regularities rather than intrinsic domain knowledge, rigorous validation remains essential to ensure that recovered dimensions reflect the intended theoretical constructs rather than spurious textual patterns.

2.1 Text Inclusion

The text-to-measure pipeline begins with selecting texts to analyze, a choice governed by two questions: what is the *population of interest*, and what *latent construct* does the researcher aim to measure? The population is the set of units—such as legislators, firms, or court cases—for which the researcher aims to generate measures, and the latent construct is the underlying attribute to be inferred about those units. A researcher should choose a text data source that supports valid measurement of the target construct across the intended units of analysis. Texts should reflect the target population, ideally spanning the full set or a representative sample of units. Moreover, the texts must exhibit enough variation in the target construct to meaningfully distinguish among units. In practice, assessing whether a potential text source is appropriate for a given measurement task requires inspecting its contents and understanding the underlying data-generating process.[6]

Once selected, these texts serve as the measurement pipeline's basic inputs: individual texts are termed *documents*, and a *corpus* is a collection of documents. *Text inclusion* is the process of determining which texts from a corpus enter the text-to-measure pipeline. Some measurement tasks seek to characterize an actor using the full set of texts they produce. If the goal is to measure hostility in public speeches by state leaders, for example, then all speeches are relevant to measurement. Because the construct is defined over the full range of the actor's communication, every speech is a draw from the population of interest. Other tasks involve measuring a property of text within a specific topic or context. For example, measuring partisan hostility entails isolating hostile language directed at the opposing party. Here, the latent construct is conditional—defined over a subset of the full corpus.

When a researcher's latent construct is defined over only a subset of the corpus under study, valid measurement of that construct depends in part on *text relevance*: the extent to which texts carry variation that relates to the target dimension. The inclusion of non-relevant text introduces two distinct measurement risks, each of which threatens construct validity. First, because low-supervision scaling infers latent dimensions from statistical regularities in text, an excess of non-relevant content can introduce systematic textual variation unrelated to the target construct. This non-relevant variation dilutes the construct-relevant signal, attenuating between-unit differences, thereby reducing the discriminatory power on which valid measurement depends. In the worst case, when non-relevant variation dominates the corpus, it can distort dimension recovery, such that the resulting measures capture a latent dimension that does not align with the researcher's intended construct.

Grimmer and Stewart (2013) illustrate the risk of measurement distortion by applying Wordfish, an unsupervised scaling model (See Section 2.3.1), to U.S.

[6] Researchers should also consider sources of systematic bias and ethical or privacy concerns in selecting their sources for text data. Grimmer et al. (2022) provides a more systematic treatment of text discovery strategies and the attendant risks of bias.

Senate press releases. Rather than recovering the target construct of left-right ideology, the estimated latent dimension primarily separated credit-claiming from position-taking, as ideological content accounted for comparatively little systematic textual variation in their corpus. Lauderdale and Herzog (2016) mitigate the risk of measurement dilution by restricting their corpus of legislative speeches to debates within a specific policy area before transforming the texts into numerical representations and scaling them to measure political disagreement. The authors note that speeches on unrelated topics introduce noise, weakening the discriminating power of construct-relevant variation across units.

Most broadly, the inclusion of non-relevant text in the text-to-measure pipeline can, for some applications, yield a scaled position when no measure should be assigned. Missing scores are often appropriate when the latent construct is a stance toward a specific political object—such as a leader, event, election, institution, court case, or scandal—because silence provides no direct textual evidence of the actor's position. In such settings, supplying non-relevant text to the pipeline risks translating general rhetoric into an inferred stance toward an object the speaker never actually addresses.

2.1.1 Text relevance as a measurement decision

The distribution of construct-relevant text—both within and across documents—varies with the corpus and the latent construct under study. In some applications, documents are internally coherent with respect to the construct, so document-level filtering can be used to retain only relevant documents. In other settings, documents contain a heterogeneous mix of construct-relevant and non-relevant content, making it more difficult to isolate relevant text. United Nations General Assembly speeches exemplify this challenge because they often embed construct-relevant foreign policy positions within material that is not relevant, such as diplomatic formalities or procedural matters. In such cases, researchers face a further choice: they can either employ documents in their original form or extract smaller text segments—such as sections, paragraphs, sentences, or clauses—that carry construct-relevant signal.

This choice entails a trade-off. Natural documents preserve *discursive context*—the rhetorical and argumentative environment in which relevant content appears—but may dilute measurement when large portions of the text are non-relevant. Restricting the pipeline to relevant document segments reduces this potential for noise, but it can also alter the structure of the textual signal. As we discuss further in Section 2.2, this matters because low-supervision text scaling derives positional information from the distributional structure of text, and those patterns may shift when words are observed in isolation rather than in their original discursive context. Accordingly, segmentation is most appropriate when the gain in construct-relevant signal from removing non-relevant content outweighs the loss of original discursive context.

If a researcher deems segmentation necessary, determining how to partition

documents is itself a measurement decision. A practical heuristic is to identify the smallest text segment that clearly expresses the target construct while retaining sufficient discursive context. In practice, this typically entails extracting relevant text in segments large enough to preserve coherence. Parsing documents too finely can undermine measurement by reducing the information content of each segment and obscuring the text's substantive meaning.[7]

Ultimately, the appropriate level of document segmentation depends on the source material, the target construct, and the data requirements of the chosen scaling approach. Because there is rarely a uniquely correct threshold, researchers should treat segmentation as a design choice that must be theoretically and empirically justified. At a minimum, this requires substantive familiarity with the source texts and a clear account of where within documents the relevant signal is likely to appear.

2.1.2 Approaches for filtering relevant text

In applications where all documents in a corpus pertain to a target construct, no filtering is required, and the researcher can proceed to the next stage of the text-to-measure pipeline. Conversely, when relevance filtering is deemed necessary, researchers should next consider how to identify and extract relevant documents or sub-document segments from their corpus. Common approaches for determining text relevance differ in the amount of human judgment they require and the *precision-recall trade-offs* they entail. We outline strategies for determining text relevance below, emphasizing the forms of classification error each approach is most likely to produce:

Definitional Concept: Precision-Recall Trade-Off

Precision and recall are complementary metrics for evaluating the performance of a classification or retrieval system. Precision is the proportion of selected documents that are truly relevant, and recall is the proportion of truly relevant documents that are selected.

In practice, these metrics exist in tension: tightening selection criteria increases precision at the expense of recall, while loosening criteria identifies more relevant documents but admits noise. The F_1 score summarizes this trade-off as the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

[7]The Comparative Manifesto Project's use of "quasi-sentences"—clauses expressing a single political idea—illustrates this concern. The boundaries between quasi-sentences have no natural delimiter, making the unit definition itself a source of arbitrary variation imposed on, rather than derived from, the distributional structure of the text (Däubler, Benoit, Mikhaylov, & Laver, 2012).

1. **Metadata-based filtering** restricts the corpus using document-level attributes, such as speaker identity, institutional setting, date, or other metadata. For example, a researcher interested in legislative debate on immigration policy might subset congressional records by committee assignment or bill topic code. This approach is straightforward and reproducible, but it relies on the accuracy and granularity of the metadata scheme. When categorization is coarse or inconsistently applied, subsetting may include substantial off-topic material (low precision) or inadvertently exclude relevant text that falls outside the selected categories (low recall).

2. **Dictionary filtering** retains texts containing term(s) associated with the target construct. This strategy is sensitive to the choice of search terms: narrow dictionaries risk excluding relevant texts (low recall), whereas broad dictionaries may capture uses of target terms in unrelated contexts (low precision). Importantly, inductively selecting keywords from the same corpus used for scaling can overfit the inclusion rule to regularities specific to the document sample from which the term list was derived. At a minimum, researchers should validate dictionary coverage against a hand-coded sample of text units not used in dictionary construction.[8]

3. **Classifier-based selection** trains a supervised model to distinguish construct-relevant from non-relevant text using hand-labeled *training data*—texts manually coded by human annotators and used to learn the relationship between textual features and category membership. Because supervised classifiers learn decision rules over a richer set of textual features, they can capture a wider range of expressions associated with the target construct than dictionary filtering. Classifier performance is typically evaluated using precision, recall, and the F_1 score. These metrics should be assessed on *test data*—hand-labeled texts excluded from training and model selection—to provide an unbiased estimate of generalization performance. For a review of the text-to-measure pipeline associated with supervised classification, see Park and Montgomery (2025).[9]

4. **Manual annotation**, in which human coders classify each document or sub-document segment as relevant or non-relevant, provides the most directly interpretable assessment of text relevance. However, it is time- and labor-intensive, difficult to scale to large corpora, and susceptible to inter-coder disagreement, especially when construct boundaries are ambiguous or abstract. Researchers using manual annotation should report inter-coder reliability statistics to assess agreement beyond chance and provide clear,

[8] Alternatively, computer-assisted approaches to keyword selection can mitigate concerns about ad hoc term selection by combining supervised classification with automated term discovery (e.g., King, Lam, and Roberts 2017).

[9] Researchers using large language models in place of trained supervised classifiers should attend to concerns about stability and opacity (see Barrie, Palmer, and Spirling 2025 and Egami, Hinck, Stewart, and Wei 2024) and should validate performance using the similar strategies described here.

reproducible coding guidelines.

No single approach to identifying relevant documents or sub-document segments dominates across all substantive applications; the appropriate strategy depends on corpus size, metadata availability, and the nature of the target construct under study. What remains constant is the obligation to be transparent about text inclusion decisions and to validate their consequences. Researchers should document their filtering procedure in sufficient detail to permit replication and report relevant performance metrics. When multiple strategies are available, combining approaches (e.g., metadata filtering followed by classifier-based refinement) can exploit the complementary strengths of each method.

2.1.3 Aligning units of analysis with document boundaries

Inclusion decisions determine what textual content enters the text-to-measure pipeline, but the texts selected for scaling do not always align with the units the researcher ultimately seeks to characterize. When the goal is to measure a latent construct at the level of individual documents, each document serves as both input to the measurement pipeline and the unit to which the resulting measure is attributed. In other applications, however, the quantity of interest is an attribute of a political actor—such as a legislator, party, or institution—rather than of any single document. A study of media slant, for instance, might use a corpus of thousands of New York Times articles to characterize that newspaper's editorial tendency. In such settings, individual texts function as repeated observations of a latent attribute defined at a higher unit of aggregation. Documents are observed manifestations of the actor's underlying latent position, and the researcher's goal is to infer the actor's underlying position from the set of texts.

Researchers must decide how to map textual inputs onto unit-level measures. This mapping is implemented via aggregation, though it can occur at different stages of the pipeline. *Pre-computation aggregation* combines raw texts before representation construction or scaling. Multiple documents or sub-document segments are concatenated into a single composite text for each target unit. These composite texts are then used to construct representations and derive measures at the level of the target unit. *Post-computation aggregation*, by contrast, operates on quantities produced later in the pipeline and aggregates them to the target-unit level.[10]

These aggregation strategies rest on different assumptions about the relationship between multiple texts and a unit-level latent position. Pre-computation aggregation assumes that texts belonging to the same unit can be concatenated into a composite document without materially altering the distributional properties relevant to measurement. In practice, however, concatenating otherwise

[10]There is a mathematically important distinction between aggregating intermediate text representations and aggregating scaled measures. Because low-supervision scaling procedures frequently involve non-linear transformations, these two approaches are not equivalent and can yield substantively different actor-level measures.

independent texts places language from distinct contexts into a shared textual environment, potentially changing word co-occurrence patterns, relative term frequencies, topic mixtures, and other linguistic regularities. The consequence is therefore not merely a repackaging of the same content, but a transformation of the distributional structure on which measurement depends.

Post-computation aggregation treats each document as a distinct pipeline input. This approach is most appropriate when the construct-relevant signal varies across a unit's documents and when preserving each document's distributional structure is important for valid representation. The aggregation problem then moves downstream: the researcher must still decide how to combine document-level representations or scaled positions into a single unit-level measure. We return to considerations for post-computation aggregation in Section 2.3.4.

2.1.4 Looking ahead: text inclusion in campaign platforms

Figure 4 consolidates the preceding discussion into a single decision framework for text inclusion in low-supervision scaling. It traces the points at which researchers must (i) define the texts to be scaled, (ii) choose a relevance criterion for retaining text, and (iii) determine whether and how to aggregate texts or downstream outputs to match the unit of analysis. By making these decisions explicit, we demonstrate that text inclusion is not a single binary decision but a sequence of linked design choices that jointly determine the scope of construct-relevant text passed to the text-to-measure pipeline.

Turning to the illustrative example in this Element makes the stakes of these decisions concrete. Campaign platforms encompass numerous issue areas within a single document, meaning that the quantity of text devoted to any given issue is small relative to the total volume of content. The textual variation relevant to a candidate's left-right position on any single issue may also be dispersed across both the document as a whole and its constituent parts.

In Section 3.3, we demonstrate how text aggregation decisions shape downstream measures of congressional candidates' issue-specific left-right positions. Specifically, we generate scaled positions using text defined at three levels of granularity: full campaign platforms, issue-relevant platform points, and issue-relevant paragraphs. Our results show that dilution and distortion often threaten construct validity and, critically, that these threats persist across scaling approaches and across different levels of researcher supervision. Taken together, these findings suggest a practical guideline: where feasible, researchers should subset texts to the most construct-relevant units; doing so more consistently places texts at their appropriate positions on a target latent dimension.

2.2 Text Representation

Once the researcher has determined which texts to supply to the measurement pipeline, the next decision concerns how to represent those texts numerically.

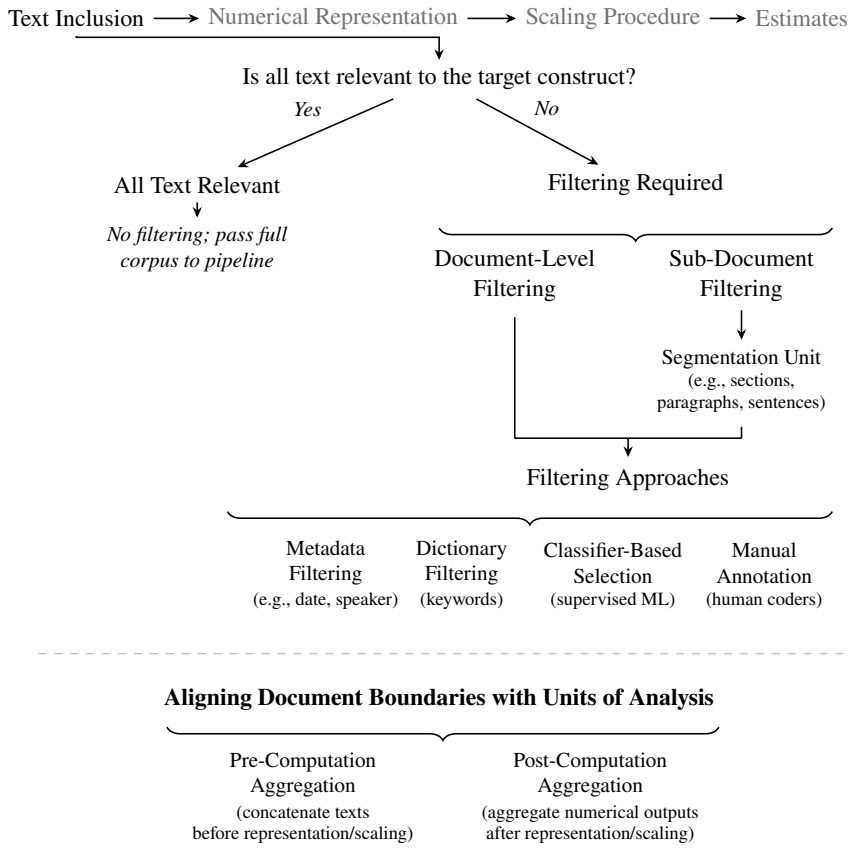


Figure 4 Decision framework for text inclusion in low-supervision scaling

Text representation refers to the process of converting texts into numerical inputs from which scaled positions can be inferred. Henceforth, we refer to all individual texts supplied to the text-to-measure pipeline as *documents*. As discussed in Section 2.1.1, these need not be natural documents: they may instead be sub-document segments (e.g., sections or paragraphs) or composite texts formed by aggregating multiple texts for units of interest (e.g., all speeches delivered by a legislator within a parliamentary session).

Representation is a consequential design choice because it determines which aspects of language—such as word frequency, word order, or contextual meaning—are encoded numerically and which are downweighted or discarded. In doing so, it shapes what textual information contributes to latent measurement. This section examines the two dominant approaches to text representation: *bag-of-words* and *embedding*-based representations. In a bag-of-words representation, each document is represented as a vector over a chosen vocabulary,

Table 2 Properties of text representation families for low-supervision scaling

	Document-Feature Matrix	**Static Embeddings**	**Contextualized Embeddings**
Unit of representation	Sparse vector over a defined feature set (usually unigram tokens; can include *n*-grams)	Dense vectors for word types (one vector per type) in learned space	Dense token-in-context vectors (one per token position in an input sequence)
Measurement signal	Variation in feature prevalence and co-variation across rows (documents)	Distributional geometry from word-type co-occurrence regularities: proximity supports semantic relatedness	Compositional context (word order; syntax); token meaning depends on surrounding tokens in the input sequence
Discarded Variation	Discards word order and syntactic structure; local context can be partially recovered with *n*-grams	Collapses polysemy; most often insensitive to compositional operators (negation/-contrast)	Limited by maximum input sequence length; tokens in different segments cannot condition each other, creates context cutoffs and boundary artifacts
Dominant basis of textual discrimination	Lexical: positions are identified by discriminating terms; most plausible with stable shared vocab and when term direction is consistent	Semantic: similar positions expressed with varied vocabularies; embeddings can map paraphrases to nearby regions via distributional similarity.	Semantic + Rhetorical: position also depends on sequence-level composition (negation, contrast, attribution, hedging, emphasis)

with each dimension corresponding to a feature—typically a unique word—and recording that feature's value in the document, such as its presence or frequency. Embedding-based representations map text units to dense vectors whose parameters are learned from data rather than indexed to a vocabulary. Our aim is not to review the technical details of these approaches, but to clarify the kinds of linguistic variation they preserve and the kinds they discard.

These representations differ in several measurement-relevant respects, summarized in Table 2. We briefly characterize each before introducing the guiding principle of this section: text representation should be chosen to match the linguistic form of the textual signal that carries the target latent construct. Selecting a representation without attention to this alignment risks weakening construct validity by obscuring or attenuating the linguistic signal that encodes the target construct.

2.2.1 Bag-of-words representation

Bag-of-words representations are typically stored in a *document-feature matrix* (DFM), with rows corresponding to documents and columns to features derived from the corpus—most often word types, but also numbers or multi-word phrases (*n*-grams). Each cell records the value of a given feature in a given document, indicating whether and to what extent the corresponding term appears in text (e.g., binary presence, raw frequency, or a weighted value such as TF-IDF or log-transformed frequency).

Under a bag-of-words representation, word order and grammatical structure are discarded. Documents are thus treated as order-invariant, and construct-relevant information is assumed to lie in term prevalence rather than term

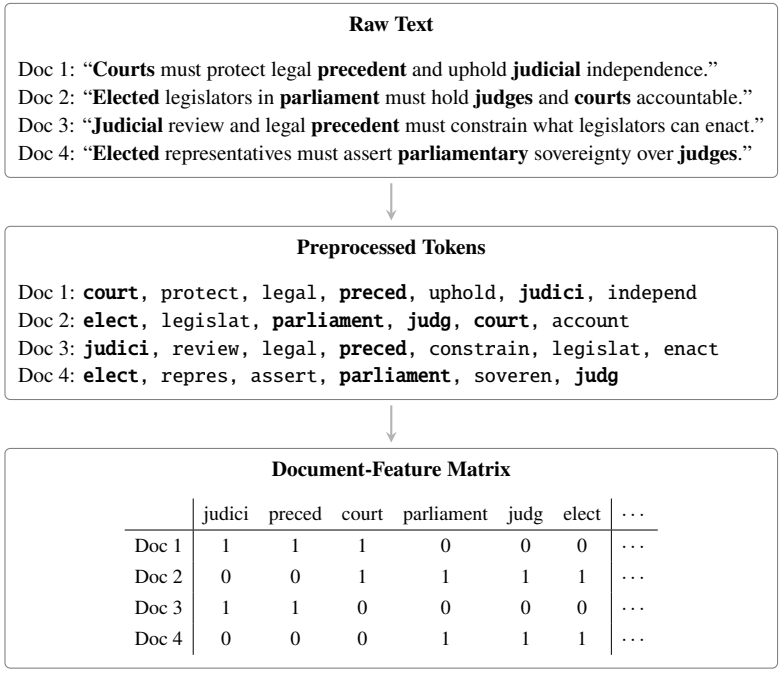


Figure 5 Converting raw text to a document-feature matrix (DFM)

arrangement. Features are likewise treated as independent, meaning that DFM representations do not directly encode semantic similarity or substitutability across terms beyond cross-document co-variation in usage. The resulting DFM is then typically passed to a scaling model that leverages this cross-document variation in term usage to recover latent dimensions. We will discuss approaches to unsupervised and weakly supervised scaling in Section 2.3.

Figure 5 summarizes the steps for converting raw text into a DFM. This process typically begins with *tokenization*, which splits raw text into discrete units. These tokens are then normalized and aggregated into terms (e.g., word types), which define the DFM's feature set. To recover some of the information lost when word order is discarded, researchers may expand the feature set beyond individual terms by defining features as multi-word spans. When features are specified as n-grams—contiguous sequences of words, such as bigrams (`judicial review`) or trigrams (`separation of powers`)—the representation preserves local word order within the n-word span. Preserving local word order comes at a cost: it sharply expands the feature space, increasing computational demands and potentially diluting construct-relevant signal with sparse or rarely observed features.

Researchers may also apply a series of preprocessing steps to raw document text before DFM construction to reduce dimensionality.[11] In some applications,

[11] Common pre-processing steps include removing stopwords (e.g., `and`, `the`, `or`), lowercasing,

such preprocessing can improve the signal-to-noise ratio by removing features unlikely to carry information about the target construct. In other applications, however, these same choices can inadvertently discard informative features. While we do not explore the downstream consequences of these forms of text preprocessing in this Element, Denny and Spirling (2018) show that seemingly routine preprocessing steps can change substantive conclusions because what constitutes noise is often construct- and task-dependent.

Once the feature set is defined, the researcher must also decide how feature prevalence will be encoded within documents. In practice, counts are often transformed or normalized (e.g., TF-IDF or log-transformed frequency), because raw frequencies can conflate construct-relevant differences in term prevalence with differences in document length. The resulting DFM is typically high-dimensional and sparse, since most documents contain only a small fraction of the defined feature set. For short documents, such as social media posts, DFMs can become especially sparse and noisy, yielding unstable measures unless the feature set is manually constrained (Lowe, 2008; Proksch & Slapin, 2009). Sparsity can be reduced by working with very large corpora or, when consistent with measurement design, by concatenating texts into composite documents before constructing the DFM (for example, see Proksch and Slapin 2010). As discussed in Section 2.1.3, however, this form of *pre-computation aggregation* can impose strong assumptions about how construct-relevant variation is distributed across the underlying texts.

Bag-of-words representations offer several advantages for low-supervision scaling. Basic validity checks are straightforward: researchers can inspect top-loading features and assess whether they map onto the targeted dimension rather than to artifacts of the corpus. These representations are also comparatively self-contained, in the sense that the feature space and feature values are defined entirely by the study corpus and the researcher's preprocessing choices. However, bag-of-words representations are particularly sensitive to researchers' decisions about text inclusion. For example, when generating a DFM, using full documents rather than relevant passages mechanically changes term distributions in ways that can dilute construct-relevant signal if off-construct content dominates, or distort measurement if the additional text introduces many terms that discriminate along a different dimension. This remains true even for weighted DFMs: because weighting is a function of corpus-level prevalence and within-document frequency, the same inclusion decision can reweight which terms are treated as informative, rather than merely adding extraneous text.

2.2.2 Embedding representation

Embedding-based approaches use machine learning models to represent texts— including individual words or full documents—as dense vectors in

stemming words to consolidate morphological variants (e.g., `judicial` and `judiciary` to `judici`), and filtering extremely rare or extremely common terms.

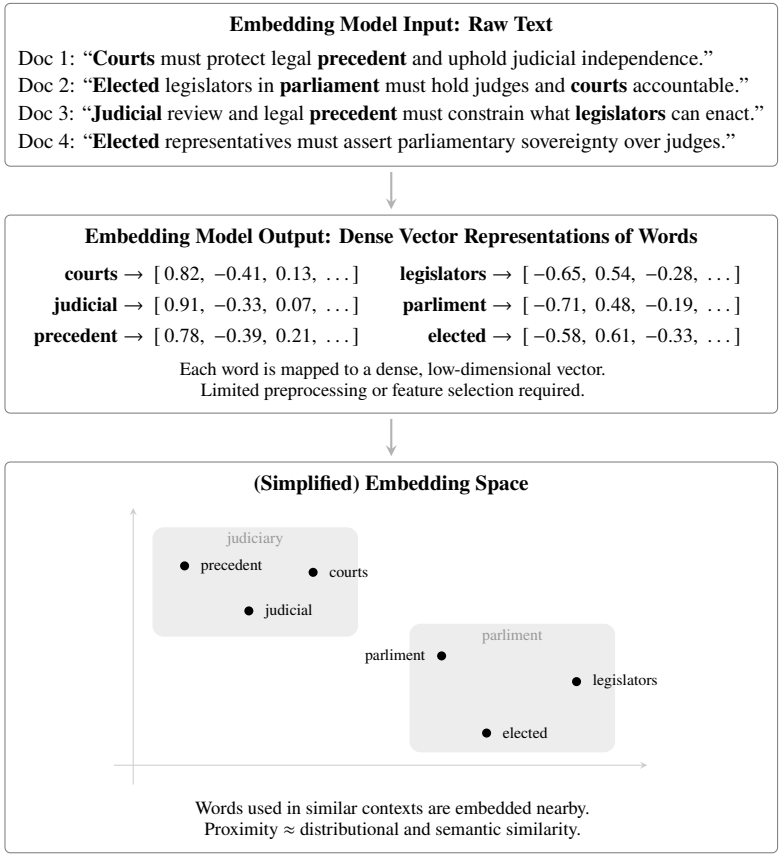


Figure 6 Converting raw text to embedding representations

a high-dimensional space, often with hundreds of dimensions. The form of these representations depends on the model's training objective, which may involve tasks such as classification or next-token prediction. In many widely used embedding models, training produces a learned space in which texts that occur in similar distributional contexts lie near one another, making geometric proximity a marker of semantic relatedness (e.g., the embedding for `court` lies closer to `judiciary` than to `parliament`). This geometry arises from systematic regularities in co-occurrence: words used in similar contexts receive similar vector representations, even when they are not synonymous (e.g., `court` lies near `judge` and `trial`). To the extent that these semantic relationships are encoded geometrically, the resulting space can support latent measurement by allowing scaled positions to be derived from similarities, distances, and projections among embeddings. Section 2.3.2 discusses strategies for computing these embedding-space quantities for scaling.

Figure 6 illustrates the geometric logic of embeddings. An embedding model

maps raw text into dense vectors (middle panel), and semantic relatedness is reflected in the relative positions of those vectors in the learned space (bottom panel). Words used in similar contexts (e.g., `courts`, `judicial`, `precedent`) are located closer together than words associated with different contexts (e.g., `parliament`, `legislators`, `elected`). Unlike DFMs, however, the dimensions of an embedding space are not directly interpretable and do not correspond to identifiable words or other explicit linguistic features. Because the mapping from text to vectors is determined by the model's learned parameters, distances and similarities are directly comparable only within a single embedding space—that is, among embeddings produced by the same model parameterization. Those parameters may come from a *pretrained* model, learned once on a large general-purpose corpus and then reused for downstream tasks, or from a *locally trained or fine-tuned* model, whose parameters are learned from scratch or updated on the study corpus.

Because semantic relationships among embeddings are defined relative to a shared vector space, model choice is especially consequential: it shapes which kinds of linguistic variation are preserved in the representation and which are attenuated. The sections that follow outline the main measurement considerations associated with two broad families of embedding models: static and contextualized. Importantly, embedding models vary both within and across these families in their architectures and training objectives. These differences shape which linguistic regularities their representations encode and, therefore, which construct-relevant variation is available for recovery. Consequently, different models applied to the same corpus can produce different embedding geometries and thus different scaled positions.

Definitional Concepts: Distributional and Semantic Similarity

Distributional similarity refers to similarity in observed usage patterns: two units are distributionally similar if they tend to occur in comparable textual environments (e.g., alongside similar neighboring words, topics, or syntactic frames). This framing follows the distributional hypothesis that systematic co-occurrence structure is informative about linguistic meaning (Firth, 1957; Harris, 1954).

Semantic similarity is a substantive claim about meaning: two units are semantically similar if they have closely related meanings or denote closely related concepts (i.e., they share core meaning components). Distributional similarity is often treated as evidence for semantic similarity, but the correspondence is imperfect and should be evaluated with external validation, such as human similarity judgments.

Static embeddings

Static embedding models assign a single vector to each input unit, yielding context-insensitive representations. Unlike DFMs, which allow researchers to

define the unit of text representation, static embedding models impose a native unit for generating embeddings. For example, Word2Vec (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013) and GloVe (Pennington, Socher, & Manning, 2014) generate embeddings for word types. Because each word type receives only one embedding, distinct uses of the same word are collapsed into a single representation: `court` as a judicial body and `court` as a basketball court are mapped to the same vector despite their different meanings.

Because the native output of these models is an embedding for each word type, producing document-level representations requires aggregating embeddings of a document's constituent words—most often by averaging, sometimes with weights—to produce a single vector for each document. This average defines a document-level embedding as a linear combination of word embeddings and provides a summary of the document's semantic content.

Averaging word embeddings can provide a strong representation for short texts, but as documents grow longer, the resulting average may become a coarser representation of semantic content (Lau & Baldwin, 2016). One alternative is to use an embedding model that directly generates document-level representations, rather than constructing them by pooling word embeddings. For instance, Doc2Vec (Le & Mikolov, 2014) learns vector representations for documents jointly with the corpus's word vectors, placing both documents and words in a shared learned space. Because document vectors occupy that common space, they can be compared directly to one another and to word vectors using cosine similarity or related distance metrics, which facilitates downstream scaling (see Section 2.3.2). The geometric intuition illustrated in Figure 5 extends to these document-level embeddings, where semantic similarity is reflected in the relative proximity of vectors in the learned space.

Because embedding models encode relational information among texts, preprocessing should generally be limited, as altering texts can remove semantic cues that embeddings are designed to preserve. For guidance on implementing static embeddings models, see Rodriguez and Spirling (2022). Although model implementation decisions are not the focus of this Element, they should be reported transparently and treated as part of the measurement design.

Contextualized embeddings

Contextualized embeddings extend this logic by producing *token-in-context* representations. Instead of assigning a single, context-invariant vector to each word type, the model generates a distinct embedding for each token occurrence conditional on the surrounding sequence. For example, `chambers` in `the judge retired to chambers` receives a different embedding than `chambers` in `the bill cleared both chambers`, because each representation is conditioned on the sequence in which the token appears. This context sensitivity allows contextualized embeddings to capture local compositional structure—including word order and syntactic relations—that is unavailable in

DFMs and only weakly reflected in static embeddings.

In contemporary text analysis, contextualized representations are most commonly generated using transformer models. Within this broad class, it is useful to distinguish between encoder models (e.g., BERT, Nomic) and decoder-only language models (e.g., GPT, LLaMA), because they generate context-dependent representations for different purposes. Encoder models produce token- or document-level outputs that can be used in low-supervision scaling as text representations. Decoder-only models likewise produce context-dependent representations, but these are used internally to support next-token prediction (i.e., language modeling) rather than being provided as outputs. In low-supervision scaling applications, decoder models are, instead, prompted to directly produce scaled positions. We will discuss decoder-only models—specifically, large language models (LLMs)—in greater detail in Section 4.

Contextualized embeddings generated by encoder models map an input token sequence to an output sequence, yielding one vector for each token position. The contextual information available to generate those representations, however, is bounded by the model's maximum sequence length. Only tokens appearing within the same input sequence can condition one another's representations. Although representations produced by the same model occupy a common learned space, the context used to generate them remains sequence-specific. This constraint is especially consequential for long documents. When a document exceeds the model's sequence limit, it must be partitioned into multiple input sequences. How that partitioning is carried out determines what contextual information is available when embeddings are generated. As a result, arbitrary sequence boundaries can have important consequences, cutting off relevant context or fragmenting coherent passages.[12]

Despite their ability to encode rich linguistic information, contextualized embeddings have other important limitations. Pretrained models inherit inductive biases from their training data and objectives that may not align with the linguistic variation relevant to the target construct. Models optimized for broad benchmark performance, for example, may emphasize topic and lexical regularities while underweighting other linguistic cues that are crucial for continuous latent measurement, such as negation, hedging, sarcasm, quotation, or coded signals. As a result, embeddings generated by certain transformer encoder models may fail to recover meaningful cross-unit variation along the dimension of interest.

These models can also pose important challenges for reproducibility and open science. Many high-performing transformer-based embedding models are only partially documented in their training corpora, filtering procedures, and optimization objectives, limiting researchers' understanding of what shapes the resulting outputs. These concerns are sharper in proprietary systems,

[12]Common steps to reduce sensitivity to segmentation decisions include partitioning texts at natural boundaries (e.g., paragraph or sentence breaks) rather than truncating mid-sentence, and introducing overlap across adjacent input sequences (e.g., a sliding-window approach) so that tokens near a boundary are embedded with some of their neighboring context.

where model versions, access terms, and hosted APIs may change over time; if embeddings shift across releases or undocumented updates, downstream measures may become unreplicable, undermining credibility and limiting meaningful comparisons across studies, time periods, and research teams. Open-weight models provide a firmer basis for reproducible research because their parameters can be downloaded, archived, and rerun locally.

2.2.3 Matching representation to construct signal

A common intuition is that text representations that encode more linguistic information—such as context, word order, and semantic relationships—are inherently superior for measuring latent constructs. In practice, the relationship is more conditional: richer representations improve measurement only when the additional information they preserve corresponds to construct-relevant variation in the corpus. To develop this point, we distinguish three main channels through which latent constructs may be signaled in text: *lexical discrimination* (word choice and prevalence), *semantic discrimination* (similar meanings expressed through non-overlapping vocabularies), and *rhetorical discrimination* (compositional structure conveyed by how words are combined in sequence). These channels are not mutually exclusive; construct-relevant variation may be expressed through multiple forms of linguistic variation simultaneously. The typology that follows is therefore intended as a conceptual tool for identifying the dominant channel(s) of construct-relevant signal in a given corpus.

Lexical discrimination

Some latent constructs are identifiable primarily through the presence, absence, or frequency of particular terms. We refer to a construct as *lexically discriminated* when differences in term prevalence account for most of the variation distinguishing units along the target latent dimension. This form of discrimination is most plausible when the construct is expressed through a well-defined vocabulary and when the directional meaning of discriminating terms is not routinely reversed by compositional devices such as negation, quotation, or sarcasm. A useful diagnostic is whether texts at different locations on the target dimension use recognizably different words in ways that remain stable across units and over time.

Consider the illustrative example motivating this Element: measuring left-right positioning on abortion within campaign platforms for U.S. House candidates. In contemporary campaign rhetoric, the words candidates use when discussing abortion are highly diagnostic of their position. Candidates on the left tend to use terms such as `choice`, `rights`, and `autonomy`, while candidates on the right more often use `life`, `unborn`, `conception`, and `heartbeat`. These lexical choices function as strategic signals that map onto the target latent dimension. Crucially, the vocabularies at each pole of the dimension are largely

distinct, and candidates only rarely draw on language from the opposing pole. When a construct is lexically discriminated in this way, positions along the target latent dimension can be inferred from systematic cross-unit differences in word usage, placing units with similar lexical profiles at similar locations.

DFMs and embedding representations both well-capture lexically discriminated constructs, insofar as both retain information about the lexical patterns that distinguish positions on the target dimension. DFMs represent those patterns directly through term occurrence and frequency, whereas embeddings encode them in distributed representations of language use. When cross-unit differences in word choice are stable and substantively meaningful, both representational families can recover the underlying positional signal.

Semantic discrimination

For other constructs, relevant variation lies primarily in semantic content rather than exact word choice. We refer to a construct as *semantically discriminated* when similar substantive positions on the target dimension are conveyed through varied, sometimes non-overlapping vocabularies. In such cases, the construct-relevant signal lies in the distributional patterns that connect different expressions to similar contexts, so texts expressing similar meanings should be located near one another on the latent dimension even when they share few surface word types. This form of discrimination is especially important when construct-relevant variation is conveyed through lexical substitution and paraphrase.

Within our running example, immigration provides a domain in which this dynamic is especially visible. On this issue, candidates may express substantively similar positions using different vocabularies: one might emphasize `amnesty`, while another stresses `legalization`. Although these terms are not perfectly synonymous, they index a common position on the target dimension in contemporary U.S. campaign rhetoric: support for granting undocumented immigrants more secure legal status. What matters for measurement in such cases is not exact lexical overlap, but whether different expressions locate texts in similar regions of semantic space.

Embedding-based representations are well suited to capturing this form of variation because they learn a continuous representational space from patterns of term co-occurrence.[13] By encoding distributional similarity, which can approximate semantic relatedness, embeddings can place texts near one another when they reflect similar positions, even if their surface vocabularies differ. This divergence in surface vocabulary poses a challenge for bag-of-words

[13]Whether these co-occurrence patterns reflect the study corpus or a general-purpose pretraining corpus depends on whether the researcher uses a locally trained, fine-tuned, or pretrained model, as discussed above. This distinction is consequential: a locally trained model captures domain-specific distributional structure, a fine-tuned model adapts general-purpose representations toward the target domain, and a pretrained model imports linguistic regularities from external data that may not align with construct-relevant variation in the study corpus.

representations, which encode texts as exact word types and therefore understate similarity when equivalent positions are expressed using different terms. As a result, semantically discriminated constructs will often be more readily recoverable from embedding-based representations than from DFMs.

Rhetorical discrimination

A third mode of textual discrimination arises when sequence-level properties of language serve as the primary signals for the target latent construct. In these cases, what matters is not simply which words appear, or even whether different words express similar meanings, but how words are combined to form propositions. Syntactic structure, modification, attribution, and other compositional features can determine whether a statement affirms, rejects, qualifies, or distances the speaker from a position. This differs from semantic discrimination, where substantively similar positions are conveyed through different vocabularies: in rhetorically discriminated constructs, the relevant variation lies less in lexical substitution than in how shared terms are assembled into meaning-bearing sequences. A related class of rhetorical discrimination is expressed through stylistic and compositional features of language, including emotional intensity, register, and linguistic complexity, conveyed through devices such as intensifiers, modality, and punctuation. In all of these cases, representations that ignore discourse and syntactic structure may blur meaningful cross-unit distinctions.

Returning to our running example of campaign platforms, healthcare provides a clear instance of rhetorical discrimination. On this issue, candidates often discuss the same policies using similar terms but convey their positions through how those terms are combined. Consider statements about the Affordable Care Act modified by `protect`, `expand`, `repeal`, or `replace`: the policy referent remains constant, but these operators fundamentally change the position expressed. Similar dependencies arise with negation: stating that Medicare for All `would not limit` patient choice conveys a different position from claiming that it `would limit` patient choice, even though the core proposition and much of the vocabulary are shared. Finally, hedging and attribution shape both the strength of a claim and its relationship to the speaker. A candidate who states that Medicare for All `may limit` patient choice differs from claiming that it `would limit choice`, and attributing that claim to `my opponent's plan` rather than `my position` alters its rhetorical force even when the underlying proposition is shared.

Contextualized embeddings are best suited to rhetorically discriminated constructs because they encode words in relation to the surrounding sequence, allowing construct-relevant variation to emerge from syntax, modification, attribution, and other compositional features of language. Bag-of-words representations are less well suited to this form of variation because they ignore word order and syntactic relations, while static embeddings remain limited because

they assign each word a single representation regardless of context. As a result, constructs that are rhetorically discriminated will more often be recovered from contextualized embeddings than either DFMs or static embeddings.

Importantly, however, not all contextualized models handle rhetorical discrimination equally well. Certain transformer encoders, such as BERT, struggle with compositional tasks, including negation detection (Ettinger, 2020). Models trained with contrastive objectives on large-scale data are better suited to capturing these distinctions, as their training explicitly encourages sensitivity to compositional semantics. We return to this point in Section 3.2.

2.2.4 Looking ahead: textual representation of campaign platforms

This section underscores that representation choice is rarely self-evident: the same corpus can support multiple plausible numerical operationalizations, and different representations trade off interpretability and transparency against semantic richness and contextual sensitivity, often alongside differences in computational burden and reproducibility. Consequently, selecting a representation is best understood as a research design decision rather than a purely technical one. It requires close familiarity with the target domain and a well-defined latent construct, along with explicit expectations about which forms of linguistic variation constitute signal versus noise.

In Section 3, we highlight that each text representation approach implies a distinct measurement pipeline. Because these pipelines differ substantially in relevant design choices, implementing multiple representation strategies in parallel to compare performance quickly becomes infeasible. We therefore treat representation choice as an ex ante design decision: researchers should use substantive expectations about the form of construct-relevant variation across units—whether lexical, semantic, or rhetorical—to select a representation family, and then assess robustness to alternative pipeline-specific implementation choices (e.g., preprocessing, input-sequence segmentation, or embedding pooling) rather than across fundamentally different workflows.

2.3 Scaling Procedure

The preceding sections examine which documents enter the text-to-measure pipeline and how included texts are transformed into numerical representations. This section turns to the final stage: *scaling* those representations onto a latent dimension intended to capture the researcher's target construct. Scaling specifies (i) how each numerical representation is mapped to a scalar position on a latent axis, which serves as an estimate of the true position π, and (ii) how that axis is *identified*. Identification involves *normalization*, which fixes the origin and units of the scale, and *orientation*, which fixes the direction of the axis (e.g., which end is high rather than low). Together, these choices make scaled positions interpretable as measures of the target construct and comparable across units.

We focus on low-supervision scaling methods that recover a latent dimension from numerical representations of unlabeled texts. Within this class of methods, we distinguish between unsupervised and weakly supervised approaches according to the degree to which they incorporate researcher-specified constraints. Unsupervised procedures extract a dimension directly from statistical regularities in the text, without substantive steering. Weakly supervised methods incorporate sparse inputs—such as reference texts, term lists, or instructional prompts—to orient the procedure toward a target construct. These methods contrast with fully supervised approaches, which use many labeled examples to estimate a mapping from texts to target scores, shifting the measurement task from latent dimension recovery to prediction.

For the low-supervision scaling methods (m) discussed below, we denote unit i's scaled position by $\hat{\pi}_i^m$.[14] Different numerical representations warrant different scaling procedures. In bag-of-words pipelines, the input is typically a document-feature matrix (DFM), and scaling leverages cross-document variation in feature frequencies to produce one-dimensional positions. In embedding-based pipelines, the input is a set of dense vectors in a learned geometric space, and scaling assigns each vector a scalar position given by its coordinate along a chosen direction in that space. Across these disparate scaling methods, the same conceptual questions recur: which encoded variation is relevant to recovering the target latent dimension, and to what extent can the researcher steer the scaling procedure toward that dimension?

Once scaling is complete, researchers can evaluate the construct validity of scaled positions. Importantly, poor validation performance does not necessarily imply that the scaling procedure is at fault. Scaling can only exploit the information encoded in the text representations it receives, and those representations are shaped by upstream choices about text inclusion. This interdependence creates a diagnostic challenge intrinsic to low-supervision measurement: validity is most naturally evaluated on the final scaled positions, but poor performance may reflect measurement failures at earlier stages of the pipeline. We return to validation in Section 3.

In the sections that follow, we outline several unsupervised and weakly supervised scaling strategies. We then discuss how to orient and normalize the resulting measures so that scaled positions are substantively interpretable. Throughout, we emphasize the discretionary choices that accumulate within each approach and highlight how these *researcher degrees of freedom* differ across unsupervised and weakly supervised methods.

2.3.1 Unsupervised approaches

Unsupervised scaling locates documents on a latent dimension inferred directly from systematic variation in numerical text representations, without

[14] We index $\hat{\pi}$ by m to denote the fact that different scaling methods will produce different positions, even for the same unit i.

construct-defining researcher input to steer dimension recovery. Intuitively, this means that there is no guarantee that the recovered dimension corresponds to the researcher's target construct. In the absence of additional identification conventions, the recovered dimension also lacks a natural interpretation, as its scale and polarity are arbitrary.

For example, Wordfish (Slapin & Proksch, 2008) infers a one-dimensional axis from a document-feature matrix (DFM) by treating feature counts as draws from a Poisson distribution whose rate (λ) depends on document length, baseline feature prevalence, and the product of the latent document position and a feature-specific discrimination parameter. The model estimates document positions and feature weights that summarize a dominant pattern of variation in language use, captured by features that discriminate among documents. Formally, let X denote the document-feature matrix, where x_{ij} is the count of feature j in document i. The Wordfish algorithm recovers:

$$x_{ij} \sim \text{Poisson}(\hat{\lambda}_{ij}), \qquad \hat{\lambda}_{ij} = \exp(\hat{\alpha}_i + \hat{\psi}_j + \hat{\gamma}_j \hat{\pi}_i^{\text{WF}})$$

where α_i captures document verbosity, ψ_j captures baseline feature prevalence, γ_j is a feature-specific discrimination parameter, and π_i^{WF} is the document position on the recovered dimension. Features with large observed $|\hat{\gamma}_j|$ are highly discriminating in that their usage covaries most strongly with document position on the recovered dimension. The fitted model yields estimates, $\hat{\pi}_i^{\text{WF}}$, which serve as our final scaled positions. The scaling procedure provides no natural zero point, unit of distance, or direction for the recovered dimension; researchers must explicitly fix its location, scale, and polarity to interpret $\hat{\pi}_i^{\text{WF}}$.

Applying principal components analysis (PCA) to bag-of-words representations follows a similar logic but imposes fewer explicit assumptions about the data-generating process. PCA identifies the one-dimensional axis in feature space that explains the most cross-document variation and assigns each document a scalar position by projecting it onto that axis. Let x_i denote the ith row of the document feature matrix X, and let $\tilde{X}$ be the column-centered version of X. Let b be the unit-length vector of feature weights for the first principal component. The first principal component score for document i is:

$$\hat{\pi}_i^{\text{PCA}} = \tilde{x}_i^{\top} \hat{b}$$

The vector b is the unit-length direction that maximizes $\text{Var}_i(\tilde{x}_i^{\top} \hat{b})$ across documents, thereby defining a one-dimensional axis that maximizes cross-document variation. In practice, PCA is usually applied to pre-processed and weighted DFMs to reduce the influence of document length (e.g., TF-IDF or log-transformed frequency) and high-frequency functional words (e.g., stop words), which would otherwise dominate the variance structure and obscure substantively meaningful patterns of language use.

Column-centering implies $\hat{\pi}_i^{\text{PCA}} = 0$ for the mean document in feature

space, giving the recovered dimension a conventional origin. However, the polarity of $\hat{\pi}_i^{\text{PCA}}$ remains arbitrary, and the scale has no inherent substantive meaning because it depends on how features are weighted and normalized. The researcher must therefore supply additional identification conventions to render $\hat{\pi}_i^{\text{PCA}}$ interpretable as a measure of the target construct. For embedding-based representations that employ PCA for scaling, the same strategy applies: documents are mapped to their positions on the first principal component of the mean-centered document-embedding matrix, again treating the dimension of maximal variation as substantively meaningful.

Unsupervised scaling encompasses a broader class of methods for mapping text representations to scalar positions without introducing researcher input into the dimension-recovery process. For instance, researchers may take the leading factor from a weighted DFM via matrix factorization (e.g., latent semantic analysis), apply unsupervised dimensionality reduction to higher-dimensional document summaries (e.g., topic-proportion vectors), or convert cluster assignments into an ordered continuum by scoring documents according to how strongly they belong to one cluster versus another (e.g., using relative proximity to cluster centroids). The examples we highlight above are intended to be illustrative, not exhaustive.

The main advantage of unsupervised methods is that they require no construct-specific inputs from the researcher to steer dimension recovery, making them attractive when such input is infeasible or conceptually contentious. The principal limitation follows directly from this property: unsupervised scaling recovers the dimension that explains the most variation in the chosen representation of the corpus, which need not correspond to the researcher's target latent construct. When a corpus contains multiple systematic sources of textual variation, unsupervised scaling may recover a dimension that conflates these sources of variation or primarily reflects non-target structure in the corpus. Researchers can sometimes mitigate these risks by narrowing the corpus through selective text inclusion criteria (see Section 2.1.2). However, such interventions can also alter the statistical structure encoded in the numerical text representation by transforming the distributional properties of the text, potentially reshaping the textual content that the measurement task aims to quantify.

2.3.2 Weakly supervised approaches

Weakly supervised scaling incorporates limited but strategically chosen domain knowledge—encoded in a form the procedure can interpret—to guide recovery of a target latent construct. Approaches differ in how they convert researcher input into constraints that fix the dimension's orientation and, in some cases, its location and scale. In practice, researchers supply *anchors*—such as reference texts, curated term lists, or instructional prompts—to define the substantive direction of the recovered dimension.

Wordscores (Laver et al., 2003), for example, uses a small set of reference

documents with exogenously specified positions to anchor locations along a target dimension. It assigns each word a score based on its association with the reference documents' positions and then places each non-reference ("virgin") document on the same dimension by taking the frequency-weighted average of its constituent word scores.[15] Formally, let X denote the document-feature matrix as above, with x_{ij} the count of feature j in document i. Let p_{ij} denote the relative frequency of feature j in document i, obtained by normalizing feature counts in X by document length. Let R index the reference documents, and let A_r denote the known position assigned to reference document $r \in R$. For each feature j, let $\tilde{p}_{rj}$ be p_{rj} renormalized across reference documents so that $\sum_{r \in R} \tilde{p}_{rj} = 1$. Wordscores assigns each feature j the score:

$$\hat{s}_j = \sum_{r \in R} \tilde{p}_{rj} A_r,$$

and assigns each non-reference document i the position:

$$\hat{\pi}_i^{\text{WS}} = \sum_{j \in V_R} p_{ij} \hat{s}_j,$$

where V_R denotes the set of features observed in the reference documents. Features not in V_R have no estimated score and are omitted from the sum.[16] Reference documents position A_r orient the scale by fixing its polarity, thereby serving as anchors for dimension recovery.

In weakly supervised embedding-based approaches, researchers typically provide curated term lists that anchor the endpoints of the target dimension. Anchor sets take various forms, from antonym pairs to larger dictionaries of terms representing each pole of the construct (Garg, Schiebinger, Jurafsky, & Zou, 2018; Kozlowski, Taddy, & Evans, 2019).[17] The anchoring terms are encoded as embeddings and averaged to produce a simple semantic summary of each pole; subtracting one pole's mean embedding from the other yields a one-dimensional semantic axis that defines the target construct. The scaling procedure then assigns each document-level embedding a scalar position by measuring its alignment with that axis, a procedure called *semantic projection* (Grand et al., 2022).[18] Scaled positions $\hat{\pi}_i^{\text{SP}}$ inherit their polarity and substantive

[15] Wordscores is often characterized as *supervised* scaling because it uses labeled reference texts. We classify it as weakly supervised because supervision enters only through a small anchor set (often a handful to a few dozen documents) that fixes the dimension's orientation and units; thereafter, document scores are determined mechanically by their lexical similarity to those anchors, rather than by fitting a predictive model on many labeled examples to estimate latent positions.

[16] This reliance on the exact feature set V_R is a key vulnerability when a construct is *semantically discriminated* (see Section 2.2.3). Consequently, this method may fail to accurately scale documents that share a latent position with reference documents but do not share their exact lexicon.

[17] Boutyline and Johnston (2025) provide guidance for producing valid and reliable term lists to define a semantic axis. Reference documents can also define the poles of a semantic axis.

[18] Another potential avenue for deriving distance-based metrics from embeddings to generate

interpretation from the researcher's choice of anchoring texts.

Formally, let $f(\cdot)$ denote an embedding function that maps a text to a vector, and let $\mathbf{z}_i = f(d_i)$ be the embedding of document d_i. The researcher specifies sets of anchoring texts corresponding to the low (L) and high (H) ends of the construct, $L = \{\ell_1, \ldots, \ell_{t_L}\}$ and $H = \{h_1, \ldots, h_{t_H}\}$. Define the two poles as mean embeddings:

$$\hat{\boldsymbol{\mu}}_L = \frac{1}{t_L} \sum_{k=1}^{t_L} f(\ell_k), \qquad \hat{\boldsymbol{\mu}}_H = \frac{1}{t_H} \sum_{k=1}^{t_H} f(h_k),$$

and define the semantic axis as:

$$\hat{\mathbf{v}} = \frac{\hat{\boldsymbol{\mu}}_H - \hat{\boldsymbol{\mu}}_L}{\|\hat{\boldsymbol{\mu}}_H - \hat{\boldsymbol{\mu}}_L\|}$$

A document's position is then given by its directional alignment with $\mathbf{v}$, typically computed using cosine similarity:

$$\hat{\pi}_i^{\text{SP}} = \cos(\mathbf{z}_i, \hat{\mathbf{v}})$$

Documents with larger $\hat{\pi}_i^{\text{SP}}$ are more semantically aligned with the high-end anchors relative to the low-end anchors, and vice versa. In this way, the anchors orient the dimension by defining the poles of the semantic axis and inducing a one-dimensional scale via projection in the embedding space.

Semantic projection can be implemented in several ways. A first distinction concerns axis construction. The bipolar formulation above defines the axis by contrasting two poles, but single-pole variants position documents based on their proximity to a single anchor set, so that larger values indicate greater affinity with that pole:

$$\hat{\pi}_i^{\text{SP,unipolar}} = \cos(\mathbf{z}_i, \hat{\boldsymbol{\mu}}_H)$$

This unipolar construction is appropriate when the concept of interest is inherently one-sided, so that only a single pole provides a meaningful basis for orienting the dimension.

A second distinction concerns how similarity is operationalized. Cosine-based projection onto $\mathbf{v}$ measures alignment along a single direction, collapsing all off-axis variation. Distance-based variants instead score documents by their relative proximity to each pole in the full embedding space:

$$\hat{\pi}_i^{\text{SP,distance}} = \|\mathbf{z}_i - \hat{\boldsymbol{\mu}}_L\| - \|\mathbf{z}_i - \hat{\boldsymbol{\mu}}_H\|$$

scaled positions is provided by Duan et al. (2025).

so that larger values indicate that $\mathbf{z}_i$ lies closer to the high-end anchors than to the low-end anchors. This approach is most appropriate when variation in the embedding space is not well summarized by a single linear axis, such that a document's relative proximity to each pole is more informative than its projection onto a line connecting them.[19]

A third distinction concerns magnitude. Cosine projection normalizes by vector length, measuring only the angle between a document's embedding and the semantic axis. Two documents pointing in the same direction receive the same $\hat{\pi}_i^{\text{SP}}$ regardless of how far they sit from the origin, ignoring variation in $\|\mathbf{z}_i\|$ across documents. When embedding norms carry construct-relevant information, such as term salience and informativeness (You, 2025), researchers can retain magnitude via dot-product projection:

$$\hat{\pi}_i^{\text{SP,magnitude}} = \|\mathbf{z}_i\| \cos(\mathbf{z}_i, \hat{\mathbf{v}})$$

which scales cosine alignment by the embedding norm. In practice, however, differences in $\|\mathbf{z}_i\|$ can also reflect nuisance variation such as document length or model-specific scaling artifacts. Researchers should therefore assess whether these differences correlate with known features of the target construct before treating magnitude as substantively meaningful.

The central advantage of weakly supervised scaling is that researcher inputs can steer dimension recovery toward a target construct even when corpus-wide variation is dominated by other factors (e.g., topic, genre, or style). However, weak supervision does not eliminate risks to construct validity. Anchor texts that are imprecise, imbalanced, or contaminated by non-construct variation can steer the procedure toward a dimension that diverges from the target construct. Moreover, weakly supervised scaling can vary in its sensitivity to alternative anchor specifications across substantive applications; in some settings, scaled positions remain stable under small perturbations to the reference documents or term lists, whereas in others, such changes produce materially different scales. Researchers should therefore assess measurement sensitivity to the definition of anchors—for example, by perturbing term lists or reference texts and re-scaling—to determine whether recovered positions are robust to reasonable alternative implementations.

2.3.3 Dimension orientation and normalization

Scaling procedures assign each numerical text representation a scalar position, but these positions are not inherently interpretable. Unsupervised and weakly supervised approaches differ in which aspects of the scale are left unconstrained, but both approaches require researchers to impose conventions that render

[19] With unit-normalized embeddings, Euclidean distance is a monotone transform of cosine distance pairwise; nonetheless, 'difference of distances' and 'axis projection' can yield different rankings because they aggregate information differently.

positions meaningful and comparable. We focus on two key components of identification: orientation and normalization.

Orientation

Orientation is the convention that assigns directionality to a recovered dimension. Which pole is assigned a positive or negative sign carries no intrinsic meaning beyond facilitating substantive interpretation of measurement.[20] When employing unsupervised scaling approaches, researchers typically determine polarity by inspecting the substantive content of texts at the extremes of the recovered dimension, examining the features that most strongly distinguish high- from low-scoring texts, and using a small number of prespecified cases to identify which end of the dimension corresponds to which substantive pole. All else equal, when a target construct has been well recovered, the polarity of the dimension should be readily recognizable. If the researcher must qualitatively inspect many scaled documents to infer the substantive meaning of a pole, this may suggest—though does not confirm—that the recovered dimension is misaligned with the researcher's target construct. Ambiguity in polarity assignment can also arise when the target construct is novel or conceptually complex, even if the dimension has been reasonably well recovered.

Weakly supervised approaches fix polarity through researcher-supplied anchors. In Wordscores, the exogenous reference positions A_r fix the scale's polarity by construction. For example, if the researcher assigns higher scores to reference documents advocating policy expansion and lower scores to those advocating policy restriction, the resulting dimension inherits that orientation. Anchor-based embedding approaches similarly inherit their orientation from the semantic axis $\mathbf{v}$ defined by the contrast between pole embeddings, which points from the low-end to the high-end anchors as specified.

Importantly, determining the orientation of a recovered dimension does not guarantee that the scaled positions reflect a target construct—further validation of scaled positions is necessary. We discuss strategies for determining the construct validity of scaled positions in Section 3.1.

Normalization

Even after orientation is fixed, scaled positions may still lack an interpretable origin or unit of measurement, since both may be artifacts of the scaling procedure and subsequent transformations. Normalization refers to the choices that fix these properties, which determine how scaled positions are expressed and compared across units or corpora.

In unsupervised approaches, normalization is determined entirely by re-

[20] This arbitrariness stems from the underlying math of dimension recovery. In PCA, for example, if b is an eigenvector that defines a principal component, then $-b$ is equally valid and explains the same variance. Most software implementations use arbitrary defaults to pick a starting sign.

searcher convention. A common practice is to standardize scaled positions to have a mean of zero and unit variance within the corpus. Under this convention, the midpoint of the latent scale is defined by the average scaled position in the corpus under study, rather than by absolute neutrality with respect to the underlying construct. In many applications, the target construct has no theoretically interpretable midpoint, making mean-zero normalization a benign convention. However, when the corpus overrepresents one side of the latent dimension, positions that appear centrist on the normalized scale may not reflect a centrist position in any substantive sense. More generally, corpus-relative normalization implies that adding or removing documents can shift every unit's standardized position, even when the underlying scaled positions are unchanged, because the sample mean and variance of the corpus change. Researchers should therefore attend carefully to corpus composition, particularly when comparing positions across time periods or institutional settings with different text distributions.

Weakly supervised approaches partially fix the recovered dimension through researcher-provided anchor texts. In Wordscores, the raw positions of non-reference documents are expressed in the units defined by A_r, the known positions of reference documents that anchor the recovered dimension.[21] In anchor-based embedding methods, document positions are defined by their similarity to a semantic axis $\mathbf{v}$, as determined by the chosen embedding model and similarity metric. In practice, measures often occupy only a narrow region of the learned embedding space, so similarity scores may span a limited range. In such cases, the scores still differ across units, but those differences are compressed into a relatively tight interval. Researchers, therefore, often rescale these projections to make differences more legible and comparisons within a corpus more straightforward.

2.3.4 Considerations for producing unit-level measures

As discussed in Section 2.1, the natural unit of text (e.g., a speech, tweet, or press release) often does not coincide with the researcher's unit of interest. Section 2.1.3 introduces pre-computation aggregation, in which texts are concatenated to the unit level before entering the text-to-measure pipeline. When aggregation is instead deferred until after scaling, and the researcher's unit of interest does not align with document boundaries, unit-level measures must be constructed through post-computation aggregation. In this approach, the researcher first estimates document-level positions, $\hat{\pi}_i^m$, and then combines them into a unit-level measure using an explicit aggregation rule. The most common choice is the unit-level mean of $\hat{\pi}^m$. Alternatives include medians or trimmed means when outliers are influential, and weighted averages when

[21] It is a well-documented mathematical property of Wordscores that the estimated positions of non-reference documents mechanically compress toward the interior of the reference document space. Applied researchers typically correct for this variance attenuation by applying LBG rescaling, which restores the discriminative range of the estimates (Martin & Vanberg, 2008).

documents differ systematically in reliability or informational value, such as length. Through these forms of post-computation aggregation, the researcher can decide how much influence unusually long documents or idiosyncratic texts should have on the unit-level measure. This is especially useful when within-unit heterogeneity is itself substantively meaningful. Researchers can also use resampling procedures applied to within-unit document sets—such as bootstrapping or leave-k-out resampling—to construct uncertainty intervals around aggregated unit-level positions.

2.3.5 Looking ahead: scaling policy positions in campaign platforms

The choice between unsupervised and weakly supervised scaling involves a fundamental tradeoff. Weak supervision improves the likelihood that the recovered dimension corresponds to the target construct, but necessarily introduces additional researcher decisions about how the target construct is defined. These choices must be theoretically justified and, where possible, subjected to robustness checks to ensure measurements are not overly sensitive to researcher degrees of freedom. Unsupervised scaling minimizes these ex ante assumptions, but offers no guarantee that the recovered dimension corresponds to the target construct rather than to some other axis of variation in the text.

In Section 3.4, we show that the relative advantages of unsupervised and weakly supervised approaches depend on both the measurement task and upstream decisions about text inclusion. When the target construct is already a dominant axis of variation in the text, unsupervised approaches can perform comparably to weakly supervised methods. However, this pattern is conditional on the scope of text included in the measurement pipeline. We apply the same scaling task to texts with varying amounts of construct-relevant material and show that, as noise increases, the construct validity of positions produced by unsupervised scaling deteriorates, whereas those produced by weakly supervised methods are less affected. Taken together, these findings suggest practical guidance: unsupervised and weakly supervised approaches can perform similarly when (i) scaled texts are construct-relevant, and (ii) the target construct is the dominant source of variation in this text. Conversely, weak supervision is preferable when the texts are noisier and the construct-relevant signal is less dominant.

3 Practical Guidance for Pipeline Design

Low-supervision scaling involves three principal design choices: (i) which texts enter the measurement pipeline, (ii) how these texts are numerically represented, and (iii) what form of researcher supervision is incorporated into the scaling procedure. Together, these cross-cutting choices generate a large space of possible end-to-end pipeline configurations, each capable of producing different scaled positions for units of interest from the same underlying corpus.

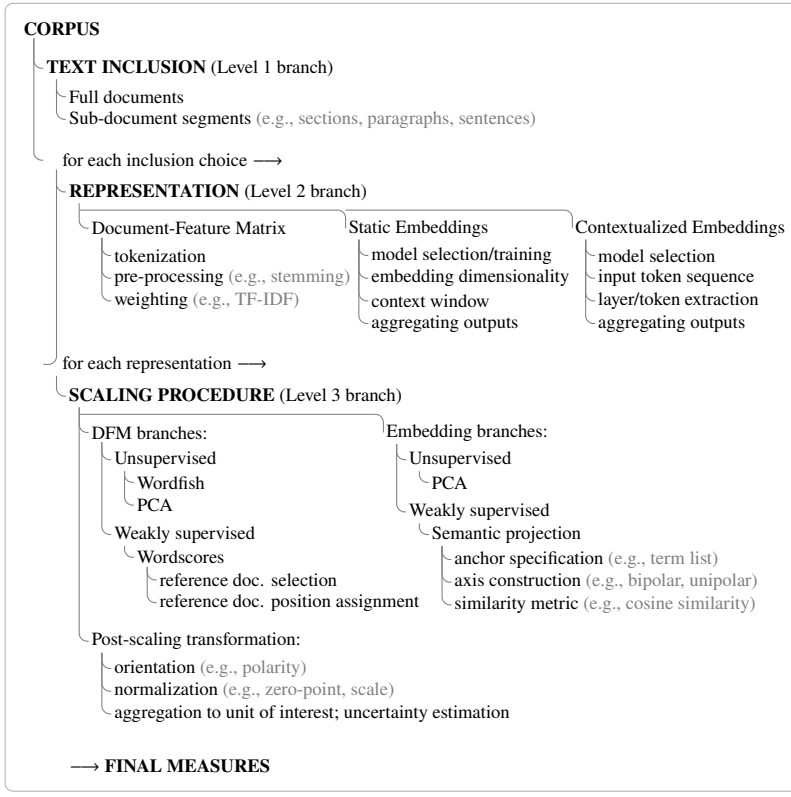


Figure 7 Example configurations of low-supervision scaling pipelines

It is unlikely that all configurations will recover the researcher's intended latent dimension equally well, and some will fail outright. Variation in pipeline performance means that choosing among configurations is itself a consequential measurement decision. Yet systematic guidance for navigating this large design space—for selecting, evaluating, and justifying a particular configuration for a given measurement task—remains limited.

Figure 7 maps a range of end-to-end pipeline configurations by organizing the principal design branches along with selected implementation decisions that are nested within them. Critically, these configurations are not assembled from interchangeable components. Different numerical representations of text can require distinct end-to-end pipelines to produce scaled positions. For example, preprocessing decisions appropriate for constructing DFM-based representations are generally inappropriate for generating embeddings (see Section 2.2). Further variation also exists within representational families, as different embedding models impose different pipeline requirements for producing representations.

Scaling is also non-interchangeable, as the procedures used to map texts onto a latent dimension are representation-dependent (see Section 2.3).

This lack of interchangeability creates a practical problem for identifying a measurement pipeline that faithfully recovers a researcher's target latent construct. In principle, exhaustive comparison of pipelines across the design space could help researchers isolate the effects of individual design choices to identify the best-performing configuration. In practice, however, even the simplified set of alternatives shown in Figure 7 yields more pipeline configurations than any researcher could reasonably implement and compare.

We therefore recommend that researchers approach low-supervision scaling as a *constrained search* over pipeline configurations. Rather than treating all possible pipelines as equally viable, researchers should begin by justifying a narrower set of plausible configurations suited to their measurement task and then compare alternative implementations within that subset. In this section, we systematically compare a broad set of pipeline configurations to generate practical guidance for narrowing the constrained search. By identifying regularities in measurement performance across pipelines, we aim to (i) prune branches of the pipeline tree that consistently yield construct-misaligned or otherwise poor-performing measures and (ii) clarify the considerations researchers can use to identify plausible setups for their own measurement tasks.

To ground these comparisons, we construct more than one hundred measurement pipelines across distinct scaling tasks. These configurations vary in the level of text inclusion, the form of text representation, and the degree of researcher supervision during dimension recovery. Specifically, we construct issue-specific measures of congressional candidates' left-right positioning in four issue domains that correspond to salient dimensions of left-right conflict in U.S. politics: abortion (pro-life vs. pro-choice), guns (access vs. restriction), healthcare (government intervention vs. free-market privatization), and immigration (inclusive vs. exclusive). Section 1.1 provides a detailed account of the motivation for this measurement task and the data used to implement it.

Our goal is not to identify a universally "best" pipeline configuration, since any generalization is likely to be qualified by important exceptions. Instead, we focus on the principal cross-cutting design choices that arise in all low-supervision scaling pipelines. Our aim is to identify patterns that can guide the selection of plausible pipeline configurations across different substantive applications, not to resolve every implementation decision within a chosen configuration. Following our constrained-search framework for pipeline design, researchers should examine the sensitivity of their preferred pipeline configuration to within-pipeline implementation choices—such as preprocessing, model tuning, or anchor specification—to ensure that the resulting measures are not driven by any particular implementation decision.

In the sections that follow, we first describe the principal design levers we vary across text inclusion, representation, and scaling. We then introduce our approach to validating scaled positions in low-supervision measurement. Finally, we apply this validation framework to empirically compare the construct validity

of scaled positions across measurement tasks and pipeline configurations. We use the resulting evidence to identify recurring patterns and offer guidance for researchers navigating their own constrained search over pipeline configurations.

Text inclusion

Campaign platforms contain natural textual boundaries at three levels: the full platform document, individual platform points, and the paragraphs within those sections. These boundaries provide a principled way to vary text inclusion within our pipeline configurations while holding the broader measurement task fixed. Moving from full platforms to platform points to paragraphs progressively narrows inclusion. This allows us to systematically assess whether excluding construct-misaligned content improves construct recovery, has little effect, or instead discards useful compositional context for scaling positions. To construct measures of left-right issue positioning under each of these three text-inclusion regimes, we proceed as follows:

1. **Paragraph:** Measures are produced from construct-relevant texts identified at the paragraph level. Each natural paragraph from a candidate's campaign platform is classified as either relevant or non-relevant to the target issue-specific dimension of left-right conflict. Paragraphs are coded as relevant only when they directly engage that dimension; text that is only topically related is coded as non-relevant (e.g., a paragraph on veterans' mental health services within a healthcare section, or a passage on human trafficking within an immigration section). We train a separate supervised classifier for each issue domain to implement this relevance filtering; full details are provided in Appendix A1.[22]

2. **Platform Point:** Measures are produced from construct-relevant platform points, where a platform point is defined as all candidate-authored text appearing under a single section heading (see Section 1.1.1). Each platform point is classified as construct-relevant if it contains at least one paragraph identified as such by the issue-specific supervised classifiers described above.

3. **Platform:** Measures are produced from the candidate's full campaign platform, including all candidate-authored text across all issue domains. No filtering by construct relevance is applied.

When generating issue-specific measures from these different levels of text, we include content only for candidates whose platforms are identified by the trained supervised classifiers as addressing the issue domain of interest. This ensures that scaled positions are generated only for candidates who discuss a

[22]This introduces an apparent tension: an approach characterized as low-supervision still relies on a supervised classifier to determine which text enters the measurement pipeline. We treat relevance filtering as supervision for text inclusion, distinct from fully supervised scaling approaches that use supervision to locate texts along a latent dimension.

given issue. In our application, that restriction is desirable because candidates who never address an issue provide no observed textual basis for estimating an issue-specific position. Assigning scores to candidates without relevant text necessarily derives issue positions from general political rhetoric rather than from any observed discussion of the issue itself. This means that the patterns we observe across text-inclusion configurations likely understate the risks posed by extraneous text, because we systematically exclude observations with no observed issue position. If we had generated positions for these units, their scores would necessarily be driven entirely by non-issue text, thereby magnifying the distortions introduced by extraneous content.

Representation

Our pipeline comparisons span two families of text representations: bag-of-words and embedding-based. With bag-of-words representations, we construct a separate DFM for each issue area using all texts deemed relevant under the applicable text inclusion rule. For platform-point and paragraph representations, relevant texts are concatenated into a single candidate–issue–year composite document (e.g., all paragraphs relevant to immigration for Alexandria Ocasio-Cortez in 2020).[23] For platform representations, we employ complete platforms as documents. We apply a standard preprocessing routine to the texts prior to DFM construction. The resulting document-feature matrices have rows corresponding to documents (candidate–issue–year units) and columns corresponding to features, with cell values recording term frequencies. For further details regarding DFM construction, see Appendix A2.1.

Embedding-based representations differ from DFMs in that the mapping from text to numerical representation is model-specific. Given the large number of possible embedding models, we do not attempt an exhaustive comparison. Instead, we focus on four models that vary along dimensions relevant to both construct recovery and practical implementation: model architecture, model transparency, and the amount of textual context each model can encode. Full details on model specification are provided in Appendix A2.2.

1. **Doc2Vec** is a static embedding model that jointly learns vector representations for words and document identifiers (Le & Mikolov, 2014). In our Doc2Vec pipelines, these identifiers are assigned to texts relevant for measuring candidates' issue-specific left-right positions, with relevance determined by the applicable text inclusion rule. When multiple texts correspond to the same candidate–issue–year unit, they are assigned the same document identifier and therefore jointly contribute to the estimation of a shared embedding for that unit. Texts classified as non-relevant are still included in Doc2Vec training,

[23] As discussed in Section 2.1.3, this pre-aggregation alters the distributional properties of the text input. We adopt this approach because individual paragraphs and platform points are often too short to produce DFMs with sufficient cross-document feature overlap, and because the concatenated segments originate from the same underlying document.

but they are not assigned a candidate–issue–year identifier and therefore contribute only to the estimation of word embeddings. After training, the model yields a single vector for each term in the corpus vocabulary and each candidate–issue–year unit.

2. **RoBERTa** is a pretrained transformer encoder built on the same core architecture as BERT, but trained more extensively on substantially more data, yielding richer bidirectional language representations (Y. Liu et al., 2019). Relative to other BERT-style encoders, RoBERTa offers a strong balance between representational performance and computational cost (Timoneda & Vera, 2025). Only documents retained under the applicable text-inclusion rule are passed to the model. In our implementation, we generate embeddings using a Sentence Transformers version of RoBERTa, which imposes a maximum input length of 256 tokens per input sequence. As a result, most documents must be partitioned into smaller segments prior to encoding. The model then produces a fixed-length embedding for each sequence, and we aggregate these sequence-level embeddings to the candidate–issue–year level using a weighted average based on sequence length.

3. **Nomic Embed** is a pretrained transformer encoder developed by Nomic AI for long-context text embedding, with a BERT-style architecture (Nussbaum, Morris, Duderstadt, & Mulyar, 2024). Its input sequence length of 8,192 tokens reduces the need for document segmentation, enabling the model to encode contextualized linguistic information distributed across longer passages. The model's open weights and training data also enhance transparency and reproducibility. In our Nomic pipelines, only documents retained under the applicable text-inclusion rule are passed to the model. When a text exceeds the model's maximum sequence length, we segment it accordingly. Nomic Embed then produces a sequence-level embedding for each segment, which we aggregate using a weighted average to obtain a single candidate–issue–year representation for each unit.

4. **OpenAI Embeddings** refers to a proprietary pretrained embedding model designed for long-text representation, with a maximum input length of 8,192 tokens. Because the model's architecture, training data, and training objectives are not publicly documented, its internal design is less transparent than that of open-weight alternatives. In our OpenAI pipelines, we encode only texts retained under the applicable text-inclusion rule. If a text exceeds the model's maximum sequence length, we segment it as needed. We then aggregate the resulting sequence-level embeddings using a weighted average to obtain a single candidate–issue–year representation for each unit.

Scaling

We distinguish between two forms of low-supervision text scaling: unsupervised approaches that recover dimensions without researcher input, and

Table 3 Truncated Dictionary of Positional Terms for Semantic Projection

Issue Area	Term Dictionary
Abortion	**Pro-Choice:** reproductive, justice, freedom, health, care, healthcare, expand, access, . . .
	Pro-Life: life, birth, death, womb, precious, sanctity, conception, heartbeat, . . .
Guns	**Increase Restrictions:** mandatory, background, ban, national, registry, assault, . . .
	Increase Access: second, amendment, owner, abiding, right, infringed, inherent, . . .
Healthcare	**Publicly Funded:** universal, all, deny, coverage, privilege, guarantee, uninsured, . . .
	Privatize: repeal, replace, market, open, competition, choice, consumers, . . .
Immigration	**Inclusive:** undocumented, pathway, roadmap, temporary, tps, dreamers, daca, asylum, . . .
	Exclusive: enforce, rule, law, secure, build, wall, barrier, surveillance, screening, . . .

weakly supervised approaches that incorporate limited researcher input to guide dimension recovery. The scaling procedure we employ depends on the numerical representation. For DFM representations, we employ Wordfish for unsupervised scaling and Wordscore for weakly supervised scaling. To anchor target issue-specific dimensions in the Wordscore scaling model, we generate and score synthetic reference texts derived from our corpus of campaign platforms. Detailed descriptions of our implementation of Wordfish and Wordscore are provided in Appendix A3.1.

For embedding-based representations, we use PCA for unsupervised scaling and semantic projection for weakly supervised scaling. To anchor issue-specific dimensions within the semantic projection framework, we specify term lists intended to capture the left-most and right-most poles of each conflict dimension. Truncated versions of these endpoint term lists are reported in Table 3. In Doc2Vec pipelines, embeddings for the endpoint terms are learned jointly with the candidate–issue–year embeddings and used to define the semantic axis. In all other embedding pipelines, comparable endpoint representations are constructed using model-specific procedures that map the endpoint terms into the same embedding space as the candidate–issue–year representations. We then obtain candidate positions by projecting each candidate–issue–year embedding onto the resulting issue-specific semantic axis. Additional implementation details are provided in Appendix A3.2.

Taken together, these design choices yield 120 pipeline configurations: 3 text-inclusion rules $\times$ (1 DFM + 4 embedding representations) $\times$ 2 scaling approaches, for 30 configurations per issue. We now turn to the validation framework used to assess the construct validity of scaled positions produced by these pipeline configurations.

3.1 Validating Low-Supervision Pipelines

Automated text analysis transforms unstructured language into quantitative measures through a series of design choices, each of which may lead pipeline outputs to diverge from a researcher's target construct. Validation is therefore

essential to assess whether scaled measures map onto the intended construct. In supervised measurement approaches, documents labeled by annotators serve as both model training data and an internal benchmark for performance evaluation. Researchers can compare model predictions against labels in held-out test data to assess whether measures recover labeled distinctions of interest. By design, low-supervision scaling approaches lack this internal benchmark. As explained in Section 2.3, these approaches recover latent dimensions based on statistical patterns in text, rather than requiring researchers to supply large amounts of labeled data for construct recovery; this reduced reliance on researcher-supplied labels is one of the main advantages of low-supervision scaling. However, without this internal benchmark, there is no systematic basis for comparing pipeline outputs against "ground truth" labels. Establishing the performance of low-supervision measurement pipelines, therefore, depends on qualitative appraisals of measures and external benchmarks for construct validity.

There are two related but distinct ways low-supervision measurement pipelines can fail to produce construct-valid measures. First, a pipeline may recover a coherent, substantively interpretable dimension that differs from the researcher's intended construct, indicating a failure of construct recovery. Second, a pipeline may recover the target dimension while mispositioning texts along that dimension, yielding construct recovery without positional accuracy.

Definitional Concepts: Face, Convergent, and Predictive Validity

Face validity concerns whether a measure appears, upon substantive inspection, to capture the target construct to informed observers. Researchers typically assess face validity by reading texts or inspecting high-loading terms. Such assessments rely on qualitative judgment and speak to interpretability rather than accuracy.

Convergent validity concerns the degree to which a measure aligns with established, independent indicators intended to capture the same latent construct. Evidence of convergent validity is stronger when the measure exhibits theoretically consistent patterns of association—in direction, magnitude, and relative ordering—with the independent indicators.

Predictive validity concerns the extent to which a measure is associated with subsequent outcomes or behaviors that the underlying construct is theoretically expected to influence. Evidence is strongest when predictions are temporally forward-looking.

For a discussion of face validity in measurement development, see Carmines and Zeller (1979). For text-analysis-oriented discussions of validation, see Grimmer and Stewart (2013); Park and Montgomery (2025); Quinn, Monroe, Colaresi, Crespin, and Radev (2010).

Distinguishing these failure modes from genuine construct recovery requires validation evidence, but no single benchmark is dispositive. Validation in low-

supervision scaling, therefore, depends on accumulating evidence across multiple exercises bearing on construct validity. Convergent validity is supported when a scaled measure aligns closely with external benchmarks, but such alignment may also reflect shared assumptions or overlapping biases, leading to spurious agreement. Predictive validity is supported when a scaled measure relates to downstream outcomes in theoretically expected ways, but these relationships may reflect confounding, in which case predictive validity may hold even when the measure fails to recover the intended construct. Face validity is supported when inspection of qualitative elements in text reveals substantively expected patterns, yet such evidence relies on selective inspection and may not generalize beyond the examples examined. Triangulating evidence across these validation exercises does not provide certainty of construct validity, but it does allow the weaknesses of any one source of evidence to be checked against the strengths of the others. The aim is therefore not to "prove" validity, but to assemble a coherent body of evidence that measures behaves as the underlying construct theory implies. In practice, pipelines that perform well on multiple validation criteria provide more credible evidence of construct recovery than pipelines that appear strong on only one dimension.

To assess convergent validity in our measurement task, we compare scaled measures of candidates' issue-specific positions to human ratings of campaign platform texts. As Lowe and Benoit (2013) argue, "valid positional estimates from quantitative scaling, for a given dimension, should match a human reader's placement of these texts with respect to identifying relative differences along this dimension" (p. 300). Because human readers are generally better at interpreting natural language, correspondence between scaled positions and human judgments provides a meaningful benchmark for construct validity. To implement this validation exercise, three human readers independently scored a random sample of campaign texts ($n = 300$) on a five-point scale ranging from very left to very right. We then average these ratings across readers and correlate the resulting scores with our pipeline-based measures. Higher correlations provide evidence of stronger construct validity, since both the human ratings and the scaled measures are intended to recover the same latent position on a left-right issue dimension from the same underlying text.

As a test of predictive validity, we examine whether the extremity of candidates' issue-specific scaled positions predicts receipt of contributions from issue-aligned Political Action Committees (PACs). PACs are fundraising organizations that pool contributions from members and donate them to candidates. Research shows that PACs contribute to candidates for both access-oriented and ideological reasons (Fouirnaies & Hall, 2014), and that candidates may adjust their positioning to enhance their appeal to PACs (Baker, 2016). We remain agnostic about the causal direction of this relationship. For our purposes, the key expectation is that candidates whose issue positions are more closely aligned with a PAC's policy priorities are more likely to receive contributions from that group (Bonica & Li, 2021; Li, 2018). To evaluate this expectation, we estimate logistic regressions predicting whether a candidate receives a contribution from

an issue-aligned PAC as a function of the extremity of their scaled position on the corresponding issue dimension. Pipelines that recover a construct-valid measure should yield a positive, statistically significant association: greater issue-specific extremity should predict a higher probability of receiving a contribution from a positionally aligned PAC.

We use keyness as a systematic basis for assessing face validity. Keyness is a signed association score that captures how much more or less frequently a word appears in a target document group than in a reference group, producing ranked lists of words that are distinctively over- or underrepresented in the target. By comparing the language used by candidates at the left and right extremes of an issue dimension with that used by candidates with more moderate positions, these analyses indicate whether the terms that distinguish the extremes are substantively interpretable and aligned with the dimension being measured. Unlike term-based diagnostics (e.g., high-loading words), keyness does not depend on a particular text representation, making it applicable across our full range of pipeline configurations. Additional detail on the implementation of this and our other validation tasks is provided in Appendix B.

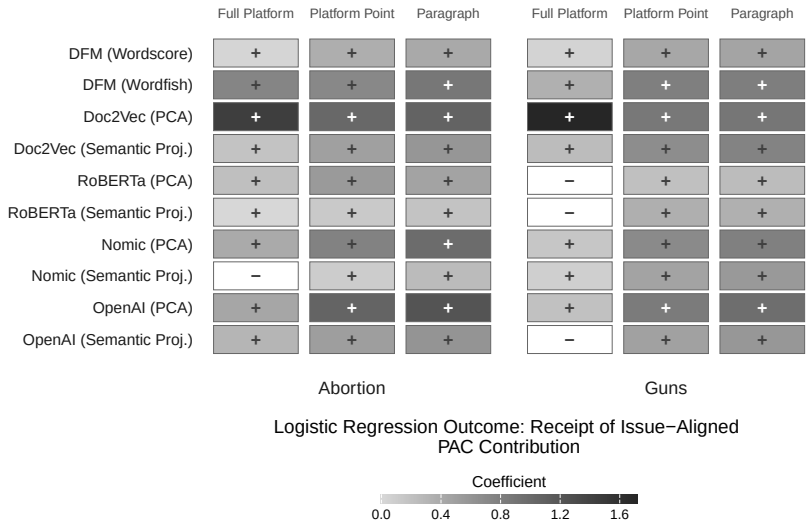
For presentational purposes, we organize our empirical comparison of construct validity across pipeline configurations around text representation as the primary branching point. Text inclusion and scaling choices are then evaluated within each representational pipeline, so that their consequences are understood relative to alternative configurations built on the same representation.

3.2 Evaluating Construct Validity: Text Representation

The construct validity of candidates' scaled left-right positions is shaped by how campaign text is represented numerically. We produce measures using two representational families: DFMs and embeddings. Moreover, we draw on multiple embedding models (Doc2Vec, RoBERTa, Nomic Embed, and OpenAI Embeddings) that capture different forms of linguistic information. Our results reveal that no representational family is uniformly superior across pipeline configurations. Instead, the approach that best recovers the target construct varies by measurement task, depending on how construct-relevant variation is expressed in the corpus. We document these patterns using the predictive, convergent, and face-validity benchmarks introduced in Section 3.1.

We first evaluate how text representation affects construct validity across pipeline configurations using our predictive validity benchmark. Figure 8 summarizes, for each pipeline configuration, the estimated relationship between candidate issue-position extremity and the likelihood of receiving a contribution from an issue-aligned PAC. The figure is organized by issue area, with each cell corresponding to a distinct combination of representation, scaling method, and text inclusion. Grayscale shading indicates the magnitude of the estimated coefficient, while the overlaid symbols denote the direction of the relationship. Recall that, if a pipeline recovers a more construct-valid measure, issue-specific

Figure 8 Predictive Validity: PAC Contributions and Candidate Issue Extremity



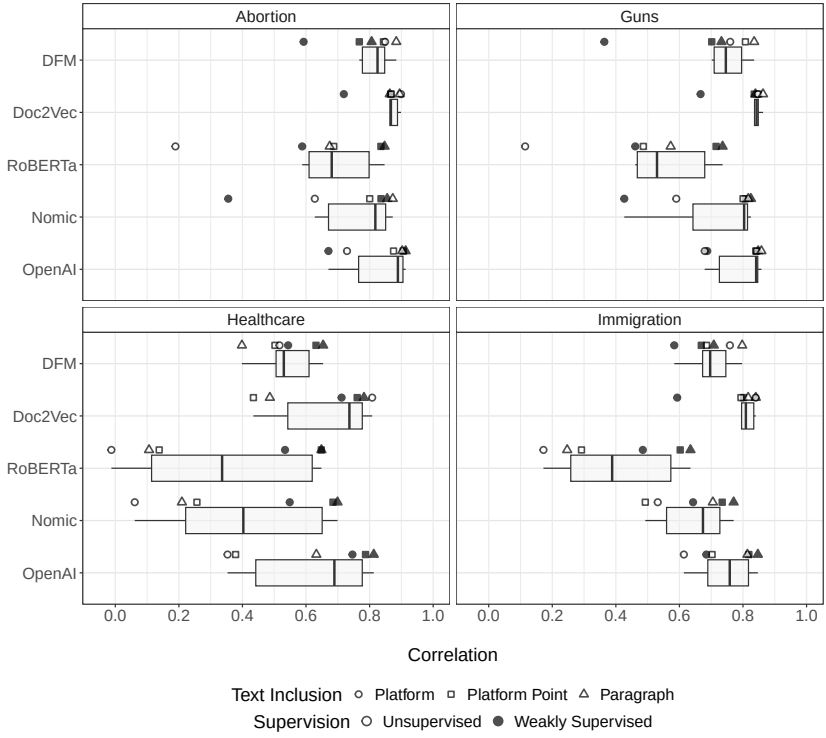
extremity should be positively and significantly associated with the probability of receiving a campaign contribution from a positionally-aligned PAC. Full logistic regression model results are presented in Appendix Section B.2.

The results in Figure 8 show that most representation-pipeline combinations yield measures that satisfy this predictive validity benchmark. Coefficients are generally positive for DFM and embedding-based pipelines, and all positive coefficients are statistically significant. These patterns indicate that candidates expressing more extreme positions on abortion or gun policy are more likely to receive contributions from positionally aligned PACs. At the same time, the figure reveals meaningful heterogeneity across representation families. DFM-based pipelines tend to produce consistently positive relationships across configurations, whereas embedding-based approaches display greater dispersion; in some configurations, they yield among the strongest predictive relationships, but in others, their associations are weaker or null.

The few instances in which the estimated relationship between position-taking extremity and PAC giving is null or negative in Figure 8 are concentrated among pipelines that scale positions from full campaign platform documents rather than more selectively filtered texts. This pattern suggests that including broader bodies of text can dilute construct-relevant signal with construct-irrelevant content, underscoring the importance of text inclusion for construct validity. We will return to this discussion in Section 3.3.

Our predictive validity benchmark necessarily covers only two issue domains, because comparable issue-aligned PACs do not exist for the remaining domains under study. This illustrates a broader limitation of external validation

Figure 9 Construct Validity: Human Text Ratings & Scaled Position Measures



benchmarks: real-world criteria are rarely clean or comprehensive, and their availability depends on the substantive context in which the construct is expressed. This dependence on imperfect and uneven external criteria underscores the value of triangulating evidence of validity across multiple benchmarks. To that end, we next examine convergent validity, which provides a broader basis for comparing pipelines within and across representation families.

Figure 9 reports, for each pipeline configuration, the correlation between scaled positions and average human-coded ratings of candidates' issue-specific left-right positioning. Each panel corresponds to an issue domain, and the rows within each panel separate correlations by representation type. Individual points denote specific pipeline configurations, where shape indicates the level of text inclusion and fill distinguishes unsupervised from weakly supervised scaling methods. Higher correlations (points farther to the right) indicate stronger convergent validity, meaning that the pipeline outputs more closely align with human judgments derived from the same underlying text. The boxplots summarize the distribution of correlations within each representation type across all pipeline configurations in that issue domain, with the median shown by the vertical line inside each box.

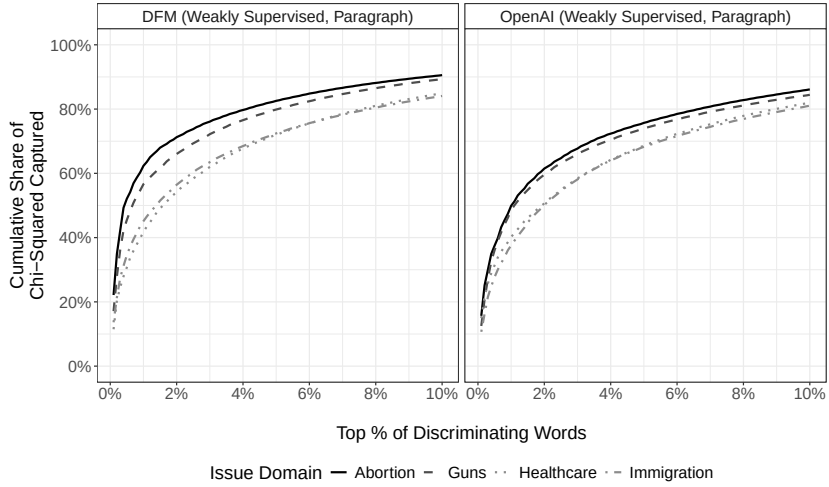
Substantial variation in convergent validity emerges across DFM- and embedding-based pipelines in Figure 9, though these differences are issue-specific rather than uniform. For abortion policy, convergent validity is consistently high across representation families, as indicated by the relatively compressed boxplots and average correlations of $\bar{r} = .79$ for DFM-based pipelines and $\bar{r} = .76$ for embedding-based pipelines. Healthcare policy, by contrast, shows a more differentiated pattern: correlations are more dispersed, the boxplots are wider, and the configurations with the highest correlations are embedding-based. Specifically, among healthcare pipeline configurations, the strongest embedding-based pipeline reaches $r = .81$ (OpenAI, Weakly Supervised, Paragraph), whereas the strongest DFM-based pipeline reaches only $r = .65$ (DFM, Weakly Supervised, Paragraph).

We argue that this variation in convergent validity reflects differences in how construct-relevant variation is linguistically signaled across issue domains, consistent with the typology of construct signals developed in Section 2.2.3. Abortion policy tends to be more lexically discriminated: candidates communicate opposing positions through relatively distinct vocabularies, making bag-of-words representations comparatively effective at recovering and ordering units along the target dimension. Position-taking on healthcare, by contrast, exhibits greater semantic and rhetorical discrimination, with positional differences conveyed through more varied vocabularies and compositional framing. Because DFMs encode term prevalence but are largely insensitive to semantic relationships and compositional structure, they are less effective at recovering positional differences expressed through these channels.

Figure 10 provides an empirical check on this interpretation by comparing patterns of lexical discrimination in the DFM (Weakly Supervised, Paragraph) and OpenAI (Weakly Supervised, Paragraph) pipelines referenced above. If a construct is lexically discriminated, then a small number of terms should reliably differentiate texts at the extreme ends of a scaled dimension. To assess this, we apply the keyness procedure described in Section 3.1 to each pipeline separately. We identify the terms most strongly associated with texts at the extremes of each issue-specific dimension as compared to all other texts.[24] We then rank terms by their chi-squared contributions—capturing the extent to which each term distinguishes extreme texts from all others—and compute the cumulative share of total chi-squared accounted for by the top 1–10% of ranked terms. Steeper curves indicate that the discriminatory signal is concentrated in a small set of terms, consistent with a more lexically discriminated issue domain. Shallower curves indicate that the discriminatory signal is more diffuse, suggesting that positional differences are conveyed through less lexically concentrated forms of variation, such as semantic relationships or compositional structure.

[24]We define target texts as those falling in the top and bottom 15% of the scaled position distribution for each pipeline, with all remaining documents serving as the reference group. This allows us to assess whether the texts assigned to the extreme tails of each pipeline's scaled distribution are distinguished by a small set of strongly discriminating terms, as expected under lexical discrimination.

Figure 10 Lexical Discrimination across Representation Type & Issue Domain



Consistent with our expectation, texts at the extreme ends of the abortion dimension are more lexically discriminated than those for healthcare. In Figure 10, the abortion curves (solid lines) rise more steeply than the healthcare curves (dotted lines), indicating that a smaller set of terms captures a larger share of the total discriminatory signal. In both plots, the top one percent of terms account for roughly half of the total chi-squared discrimination for abortion, whereas the corresponding share for healthcare is substantially lower. These patterns support our theoretical argument: in measurement tasks where a discriminatory signal is concentrated in distinctive vocabulary, DFM-based representations can match the construct recovery achieved by high-performing embedding-based alternatives. When measurement tasks are more semantically or rhetorically structured, embedding representations have clearer advantages.

These results might tempt researchers to default to embedding representations in their own pipelines, on the assumption that encoding more linguistic information yields more reliable recovery of construct-relevant signal across diverse measurement tasks. We caution against that assumption. As Figure 9 shows, several embedding-based pipelines consistently underperform DFM-based alternatives. In the immigration task, for example, average correlations across Nomic- ($\bar{r}$ = .65) and RoBERTa-based pipelines ($\bar{r}$ = .41) are appreciably lower than for DFM pipelines ($\bar{r}$ = .70), as well as other embedding-based alternatives (Doc2Vec: $\bar{r}$ = .78; OpenAI: $\bar{r}$ = .75). The boxplots show that this pattern is not driven by isolated outliers: the distributions for RoBERTa- and Nomic-based pipelines lie to the left of the DFM distribution, indicating systematically lower convergent validity across configurations. More broadly, RoBERTa-based pipelines are the weakest or near-weakest performers across all four issue domains, despite encoding richer contextual information than DFMs. This pattern makes clear that representational richness, on its own, is

not sufficient for ensuring strong construct validity in scaled positions.

One plausible explanation for this tension is that representational richness benefits measurement only when it captures linguistic information relevant to the target construct. Prior work suggests that pretrained contextual models are not uniformly superior at recovering compositional distinctions (Ettinger, 2020), and that general-purpose optimization for broad task performance does not guarantee alignment with task-specific linguistic signals (Gururangan et al., 2020). As a result, general-domain pretrained models may encode substantial variation unrelated to the target dimension, incorporating construct-irrelevant noise into the representation. In such settings, simpler representations can outperform richer alternatives by preserving construct-relevant distinctions without encoding as much irrelevant variation.

Taken together, these results show that representation choice should be guided by the form of construct-relevant variation researchers seek to recover, not by a presumption that richer representations are categorically better. We therefore begin our recommendations with principles for aligning text representation to the measurement task.

Recommendations for Choosing a Text Representation

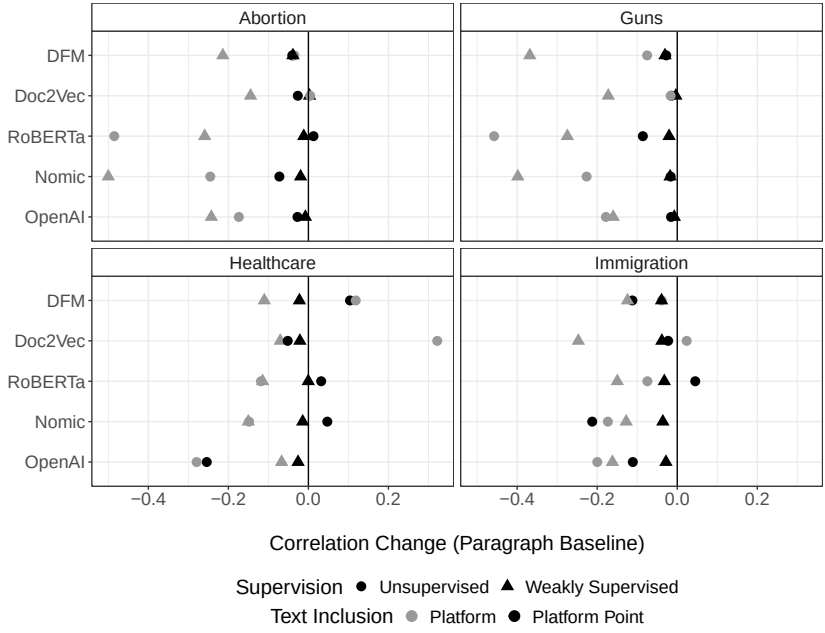
1. Representation choice should begin with an exploratory examination of the corpus. Researchers should draw on substantive knowledge and targeted reading of sample documents to assess how construct-relevant signal is expressed in text—whether differences across texts are carried by distinctive vocabularies, by shared vocabularies used in systematically different ways, or by broader patterns of framing and rhetorical structure. Because the form of construct-relevant signal is an empirical property of the corpus, it should not be assumed to generalize across applications.

2. Text representations should be chosen to match the form of signal the researcher seeks to recover. When construct-relevant differences are primarily lexical, DFMs can perform well. When they are expressed more through meaning, framing, or rhetoric, embedding-based representations may better capture the relevant variation. Even so, greater representational richness does not by itself guarantee stronger construct recovery. When possible, researchers using embeddings should compare multiple models rather than relying on a single representation.

3.3 Evaluating Construct Validity: Text Inclusion

Text inclusion determines which documents—or segments thereof—enter the text-to-measure pipeline. We vary text inclusion across three levels of

Figure 11 Convergent Validity: Changes in Correlation with Human Ratings Relative to Paragraph Baseline



granularity: paragraphs, platform points, and full campaign platforms. Each level progressively incorporates a greater share of construct-irrelevant text into the pipeline by broadening the scope of included content. Our results demonstrate that these inclusion choices systematically shape construct validity, with scaled positions derived from more construct-relevant text consistently outperforming those derived from aggregated text inputs.

Recall from Figure 8 in Section 3.2 that construct validity varies across pipeline configurations differing in text inclusion. Paragraph-level configurations most consistently recover positive, statistically significant relationships between candidate extremity and positionally aligned PAC contributions, whereas full-platform specifications yield a small number of null or negative estimates. This pattern suggests that construct validity degrades as the scope of text inclusion broadens. However, this evidence is limited: the null or negative estimates appear in only four of the twenty pipeline configurations that employ full-platform text inclusion, and the predictive validity task does not cover the healthcare and immigration domains.

To more systematically assess how text inclusion affects construct validity, Figure 11 re-expresses convergent validity relationships between human ratings and our scaled measures of candidates' left-right issue positions.[25] Rather than

[25]For the original figure, which reports raw correlations, see Figure 9 in Section 3.2.

reporting raw correlations, the figure presents changes in correlations across measurement pipelines relative to the paragraph-level text configuration, which we treat as the baseline given its expected concentration of construct-relevant content. The vertical reference line at zero marks parity with paragraph-level inclusion: points to the right indicate that pipelines with broader inclusion outperform this baseline, while points to the left indicate stronger correlations under paragraph-level inclusion, holding all other pipeline components constant. Each panel corresponds to an issue domain, with rows separated by representation types; point shading indicates text inclusion level, and shape distinguishes unsupervised from weakly supervised scaling methods.

The correlational changes in Figure 11 show that construct validity generally degrades as text inclusion expands to incorporate more construct-irrelevant content. In 70 of the 80 plotted comparisons, changes fall to the left of zero, indicating weaker construct recovery relative to the paragraph-level baseline; only three cases show notable improvements under higher levels of aggregation. Moreover, the grey points typically lie to the left of the black points, indicating deterioration as text inclusion expands from platform points to full platforms. This pattern persists across most representation types and scaling methods, indicating that, in the settings examined here, incorporating substantial amounts of construct-irrelevant text has systematic consequences for measurement.

We illustrate the substantive consequences of text inclusion for construct validity in Table 4. We report the 25 highest-keyness terms—those that most strongly discriminate texts assigned left- and right-extreme positions from more moderate positions—for two pipelines (Nomic, Weakly Supervised and DFM, Unsupervised) across varying scopes of text inclusion.[26] This test differs from our preceding keyness analysis, which compared extreme texts to all others. Here, we instead compare extreme texts to more moderate texts on the same side of the issue dimension in order to isolate the language associated with extremity, rather than broader differences between the left and right tails. Terms that are construct-irrelevant (i.e., generic political language or terms unrelated to the policy domain) are underlined. A higher proportion of underlined terms indicates that differences between extreme and moderate positions are unrelated to the target latent construct. Importantly, the texts used in the keyness analysis are restricted to paragraphs classified as relevant by the supervised issue classifier. Thus, even in the platform- and platform-point comparisons, we analyze only construct-relevant paragraphs. If we observe a larger proportion of irrelevant keyness terms, this suggests that differences between extreme and moderate scaled positions reflect weak construct-relevant discrimination, not noise induced by including longer documents with non-relevant content.

At the paragraph level, the top keyness terms for both pipelines align closely with the gun policy dimension: right-extreme texts are characterized by Second Amendment language (e.g., right, bear, arms, amendment, constitutional), while

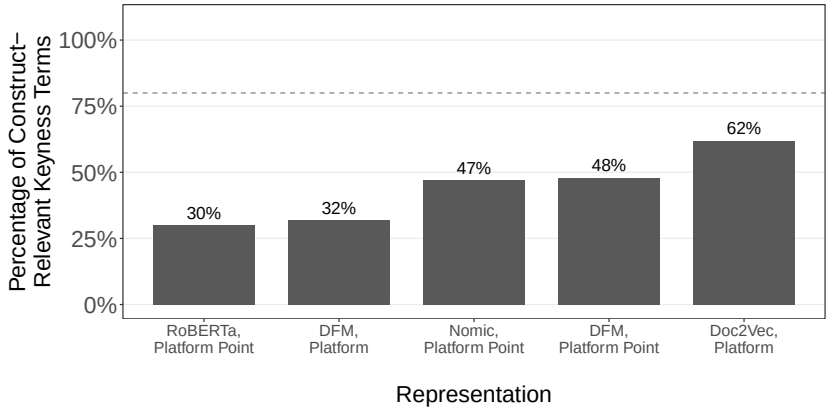
[26]We define target texts as those falling in the top 15 percent or bottom 15 percent of the scaled position distribution for each pipeline, and compare them to texts on the same side of the distribution that fall between the 50th and 85th percentiles or between the 15th and 50th percentiles, respectively.

Table 4 Top Keyness Terms for Left- and Right-Extreme Texts on Gun Policy

Pipeline Configuration	Top Keyness Terms
Nomic, Weakly Supervised *(Right-Extreme)*	**Paragraph:** right, freedoms, constitution, liberties, rights, speech, constitutional, given, bear, amendment, god, government, arms, freedom, second, fundamental, free, tyrannical, protect, religious, protects, defend, liberty, keep, liberals . . .
	Platform Point: right, constitution, freedoms, rights, amendment, bear, arms, constitutional, second, speech, free, given, government, god, liberties, fundamental, keep, freedom, liberty, tyrannical, protect, upon, defend, protects, attempts . . .
	Platform: right, amendment, second, arms, bear, rights, constitution, liberty, freedom, protect, religious, freedoms, government, defend, infringed, shall, upon, free, speech, god, keep, unconstitutional, document, values, oregon . . .
(Left-Extreme)	**Paragraph:** weapons, assault, mass, ban, high, killed, united, suicide, buy, deadliest, shootings, year, color, country, market, war, loophole, day, invest, buyback, sales, guys, politicians, semiautomatic, time . . .
	Platform Point: weapons, ban, assault, high, mass, background, tragedies, capacity, guys, ammunition, shootings, buy, killed, united, war, magazines, loophole, close, unacceptable, headlines, check, day, purchase, sales, failed . . .
	Platform: invest, americas, university, caliber, hospital, heres, iraq, built, politicians, need, average, death, start, month, third, suicides, belong, equipment, seller, partner, fewer, bad, gang, die, poverty . . .
DFM, Unsupervised *(Right-Extreme)*	**Paragraph:** right, bear, arms, infringed, shall, keep, constitution, amendment, second, defend, freedoms, fundamental, rights, speech, constitutional, liberties, preserve, always, period, upon, fight, government, firm, freedom, protects . . .
	Platform Point: right, bear, arms, infringed, keep, shall, second, amendment, defend, rights, constitution, fundamental, freedoms, constitutional, fight, protect, given, speech, god, upon, always, protected, defender, period, liberties . . .
	Platform: god, right, left, government, shall, fight, rights, amendment, second, arms, bear, away, stand, defense, infringed, heart, keep, back, grabbers, joe, restrict, liberty, given, massachusetts, speech . . .
(Left-Extreme)	**Paragraph:** violence, assault, ban, domestic, background, weapons, survivors, loophole, capacity, sexual, magazines, high, epidemic, act, checks, prevention, sale, close, universal, sales, loopholes, hr, closing, charleston . . .
	Platform Point: violence, domestic, community, sexual, prevention, programs, assault, invest, survivors, loophole, close, communities, epidemic, health, high, based, women, loopholes, services, ban, vawa, magazines, abusers, capacity, background . . .
	Platform: violence, act, survivors, prevention, communities, community, sexual, bipartisan, programs, cosponsor, hate, co, housing, historic, task, joe, intervention, bill, authored, hr, assault, based, house, sponsored, vawa . . .

left-extreme texts emphasize gun violence prevention (e.g., violence, background, checks, assault, ban). This alignment largely persists at the platform-point level. Under full-platform inclusion, however, the highest-keyness terms frequently include content that is not directly related to gun policy or that reflects general political rhetoric rather than construct-relevant content, especially among left-extreme terms, where gun-specific language gives way to broader public health, social justice, and community-oriented vocabulary that is shared across multiple

Figure 12 Face Validity: Construct-Relevant Keyness Terms in Healthcare Pipelines Outperforming Paragraph Baseline



policy domains. This shift indicates that, in full-platform configurations, the features driving discrimination between texts at adjacent left-right positions (i.e., extreme-left versus moderate-left, or extreme-right versus moderate-right) are increasingly dominated by diffuse, cross-domain rhetorical signals rather than construct-relevant policy content. This calls into question the face validity of scaled positions generated under full-platform inclusion, as texts assigned extreme positions are distinguished from their positional neighbors on the basis of content that has little to do with gun policy.

The importance of inspecting discriminative features for alignment with the target construct becomes especially apparent when applying the same face-validity exercise to pipelines that outperform the paragraph-level baseline in Figure 11. For instance, the Doc2Vec (Unsupervised, Platform) configuration for healthcare policy exhibits a positive change in correlation ($\Delta r = 0.32$) and achieves the highest performance among Doc2Vec-based pipelines ($r = 0.90$), suggesting that expanded text inclusion improves measurement in this case. However, as we demonstrate below, the corresponding discriminative terms provide little evidence of face validity.

Figure 12 plots the proportion of discriminative, high-keyness terms that are substantively aligned with the target construct for healthcare policy pipelines that outperform their paragraph-level baseline in Figure 11. For each pipeline, we identify the top 50 keyness terms associated with texts at the left and right extremes of the scaled dimension and assess whether these terms are construct-relevant.[27] We additionally include a horizontal reference line corresponding to the highest-performing healthcare pipeline (OpenAI, Weakly Supervised, Paragraph), to indicate the degree of face-valid construct alignment achieved by

[27] Non-construct-relevant terms are analogous to those underlined in Table 4, reflecting generic political or cross-domain language rather than policy-specific content.

the best-performing configuration.

Despite their improved correlations, these pipelines recover a limited share of construct-relevant terms, ranging from 30% to 62%, indicating that the terms driving separation between more extreme and more moderate texts are often not closely tied to the target construct. In practical terms, movement toward the ends of the scaled dimension is not consistently associated with more substantively diagnostic policy content. Taken together, the figure demonstrates that there is no consistent improvement in construct alignment as text inclusion expands to platform-point or full-platform representations, reinforcing our argument that broader text inclusion introduces diffuse, cross-domain signals. Furthermore, this divergence emphasizes that each validation test has its own failure modes: improvements in correlation-based metrics may be spurious, indicating increased sensitivity to broader rhetorical or stylistic signals rather than better recovery of the target construct.

The results in this section show that text inclusion meaningfully shapes construct validity across pipeline configurations. Narrower inclusion strategies consistently outperform broader ones by concentrating construct-relevant signal in the input text. When construct-irrelevant content enters the pipeline, the recovered dimension increasingly reflects general political orientation rather than domain-specific positioning—a pattern visible in both the correlational evidence and the discriminative features of scaled measures.

Recommendations for Specifying Text Inclusion

1. Text inclusion should begin with an assessment of where construct-relevant signal is expressed within documents. Researchers should use substantive knowledge and targeted reading to determine whether the construct is conveyed in focused segments (e.g., paragraphs, sections, or passages) or more diffusely across full text units.

2. Text inclusion should be specified to align with the level at which construct-relevant variation is concentrated. When the signal is localized, more narrow text inclusion can improve construct recovery by excluding irrelevant content. When the signal is more diffuse, broader inclusion may be necessary, but can introduce additional sources of variation that are not directly related to the target construct.

3. When construct-relevant text cannot be reliably isolated—for example, because topic boundaries are diffuse or metadata is unavailable—the resulting measures should be treated with additional scrutiny. In these settings, inspection of discriminative features is essential, as pipelines operating on broad, unfiltered text can achieve strong construct and predictive validity while still recovering a dimension that is poorly aligned with the target construct. We discuss strategies for mitigating these risks in Section 3.4.

3.4 Evaluating Construct Validity: Scaling Procedure

Supervision refers to the extent and form of researcher guidance used to map numerical text representations onto positions along a latent dimension. Low-supervision scaling approaches vary in the extent to which they incorporate researcher input into construct recovery, ranging from fully unsupervised dimension extraction to minimally guided procedures. Our results suggest that supervision functions as a guardrail for construct recovery. When the text is coherently aligned with the construct or the target dimension is a dominant source of variation in the corpus, additional supervision yields comparatively modest improvements. However, when representations inadequately encode the target construct, or when the construct itself is diffuse in the text—either because it is inherently nuanced or because relevant signal is diluted by substantial irrelevant content—supervision helps stabilize the mapping and maintain alignment with the intended concept.

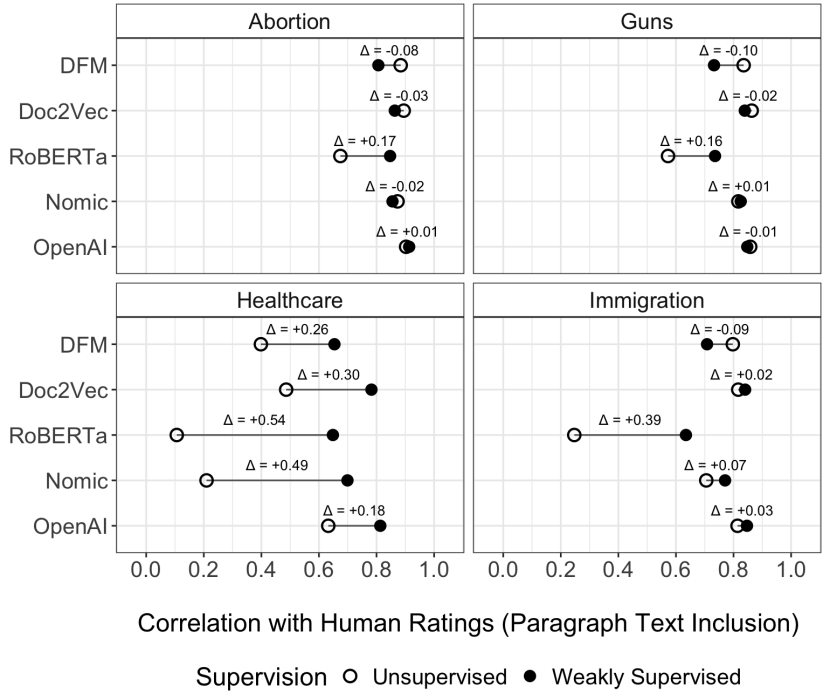
Figure 13 once again plots the relationship between human ratings and our scaled measures of candidates' left-right issue positions.[28] The figure is restricted to pipelines using paragraph-level text inclusion. We report raw correlations alongside annotations for the change in correlation (Δ) between weakly supervised and unsupervised measures, holding all other pipeline components constant. Positive values of Δ indicate that weakly supervised pipelines produce stronger correlations with human ratings than unsupervised approaches; negative values indicate weaker correlations.

The convergent validity evidence in Figure 13 shows that the relative advantage of weak supervision varies substantially across issue domains. For abortion and gun policy, the supervision gap is negligible. Across all five representation types, Δ values cluster near zero or are weakly negative, indicating that unsupervised scaling recovers the target dimension about as well as—and sometimes better than—weakly supervised alternatives. Healthcare and immigration, by contrast, exhibit larger and more consistently positive changes. In healthcare, weakly supervised pipelines outperform unsupervised ones by margins ranging from $\Delta = +0.18$ (OpenAI) to $\Delta = +0.54$ (RoBERTa), with positive gaps across all five representations. Immigration shows a similar but more heterogeneous pattern: RoBERTa-based pipelines display a substantial supervision advantage ($\Delta = +0.39$), while Doc2Vec and OpenAI pipelines exhibit near-zero differences.

Notably, the largest gains from supervision appear where unsupervised scaling performance is weakest. In healthcare, RoBERTa and Nomic—the two representations with the lowest unsupervised correlations among paragraph-based pipelines—exhibit the largest improvements ($\Delta = +0.54$ and $+0.49$, respectively). The same pattern holds for immigration; RoBERTa's unsupervised correlation is substantially lower than that of other representations, yet its weakly supervised counterpart improves to a comparable level. This pattern suggests

[28] For the original figure reporting raw correlations, see Figure 9 in Section 3.2.

Figure 13 Convergent Validity: Text Ratings & Scaled Position Measures, with Correlational Changes

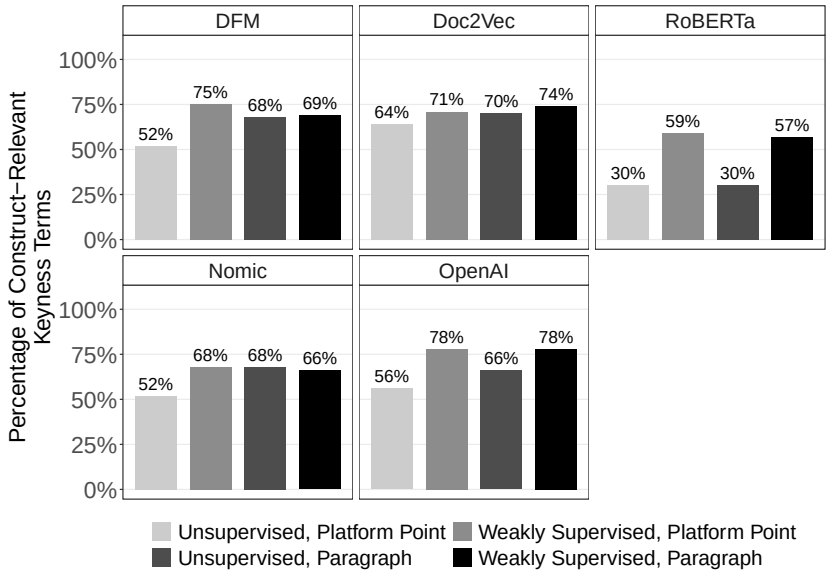


that weak supervision is most consequential as a corrective when representations capture the target construct imperfectly (see Section 3.2).

The same logic should apply when text inclusion is broad, such that substantial construct-irrelevant content enters the pipeline, creating conditions under which weak supervision can provide a similar corrective. To test this expectation, we compare changes in correlations, analogous to those shown in Figure 11, between scaled positions generated by pipelines using paragraph-level inclusion and those generated by pipelines using platform-point inclusion. Positive values of Δ indicate that weakly supervised pipelines produce stronger correlations with human ratings than unsupervised approaches; negative values indicate weaker correlations. We find that, as text inclusion broadens, the relative advantage of weak supervision generally increases.

Immigration illustrates this pattern most clearly. Under paragraph-level inclusion—where texts are filtered on the target dimension—the supervision gap is negligible for most representations: Δ for DFM, Doc2Vec, Nomic, and OpenAI all fall within ± 0.03. When text inclusion is broadened to platform points—which introduces more construct-irrelevant text— the gap in correlational changes widens appreciably. Nomic's Δ increases from +0.07 to

Figure 14 Face Validity: Construct-Relevant Keyness Terms



+0.24, OpenAI's from +0.03 to +0.12, and RoBERTa's remains substantial at +0.31. Only DFM and Doc2Vec remain near zero. This pattern is consistent with the expectation that as construct-irrelevant content enters the pipeline, unsupervised scaling is more likely to drift toward an alternative dimension, while weakly supervised methods remain anchored to the target construct.

This interpretation is further substantiated by the face-validity evidence in Figure 14, which examines the proportion of discriminative (high-keyness) terms that are substantively aligned with the target construct under different supervision regimes. For each combination of supervision and text inclusion, we identify the top 50 keyness terms associated with texts at the left and right extremes of the scaled dimension relative to more moderate texts, and assess whether these terms are directly related to immigration policy.

For paragraph-level inclusion pipelines, both unsupervised and weakly supervised pipelines produce discriminating vocabularies that are overwhelmingly construct-relevant, with about 70% of top keyness terms directly related to immigration across all representations. However, for platform-point inclusion, the unsupervised pipelines deteriorate sharply: the share of construct-relevant terms drops to 52% for DFM and Nomic, 56% for OpenAI, and 30% for RoBERTa. The terms that replace immigration-specific vocabulary are drawn from unrelated policy domains (e.g., `education, climate, housing, healthcare, lgbtq`). Weakly supervised pipelines degrade much less with broader text inclusion: construct-relevant shares remain around 70%. These patterns indicate that weak supervision does not merely improve correlational fit with human

ratings, but substantively reorients the features driving discrimination toward the intended latent dimension. In this sense, supervision serves as a constraint on feature salience, reducing the influence of construct-irrelevant variation.

Taken together, these results show that the value of supervision depends on the extent to which the target construct dominates the variation available to the scaling procedure. Unsupervised and weakly supervised approaches can perform similarly when (i) scaled texts are construct-relevant, and (ii) the target construct is the dominant source of variation in this text. However, as representations capture the target construct less effectively or text inclusion broadens, unsupervised pipelines are more likely to drift toward alternative dimensions. Weak supervision mitigates this risk, but does not eliminate the effects of poor upstream choices. Supervision should therefore be treated as one component of a well-specified pipeline, not as a substitute for careful decisions about text inclusion and representation.

Recommendations for Choosing a Scaling Procedure

1. Unsupervised scaling minimizes researcher degrees of freedom and avoids the practical burden of specifying and defending task-specific anchors. However, without researcher guidance, the recovered dimension reflects the most salient axis of variation in the representation, which may not coincide with the target construct. This risk increases when text inclusion is broad, when the representation encodes substantial construct-irrelevant variation

2. In practice, researchers may not know ex ante whether their target construct dominates the variation in the representation. A reasonable starting point is to implement an unsupervised pipeline and inspect the resulting discriminative features for substantive alignment with the target construct. If the recovered dimension is well-aligned, unsupervised scaling may be sufficient. If discriminative features indicate misalignment, researchers should introduce weak supervision to anchor recovery to the target construct.

3.5 Summary of Practical Guidance

The guidance presented here yields several high-level takeaways. First, representation choice should align with the form of construct-relevant signal in the text, whether lexical, semantic, or rhetorical. Second, whenever possible, text inclusion should be concentrated at the lowest coherent level to limit the prevalence of construct-irrelevant content. Third, supervision can provide a corrective when representation and inclusion decisions are imperfect, but it is not a substitute for a well-designed pipeline.

Within our constrained-search pipeline framework, researchers should demon-

strate that scaled positions are not unduly sensitive to any single specification choice within the chosen pipeline. For DFM-based pipelines, this may involve varying preprocessing steps, such as removing stop words or high-frequency terms, as well as employing alternative weighting schemes to adjust the contribution of common versus distinctive terms to the recovered dimension. For embedding-based pipelines, this may involve varying the context window used to generate embeddings, adopting different pooling strategies for constructing document- or unit-level representations, or implementing the pipeline with different embedding models. For pipelines that rely on weak supervision for scaling, researchers may also vary the anchor texts used to define the recovered dimension. These are only some of the many specification choices that can be varied to demonstrate pipeline robustness.

Across all low-supervision pipeline implementations, transparency and reproducibility are essential. Researchers should clearly document all consequential design choices, including how texts are selected, represented, scaled, and aggregated, along with the substantive rationale for each choice. Researchers should also evaluate scaled positions using multiple forms of validation to assess whether the recovered dimension reflects the intended construct. Because different validation exercises are informative about different aspects of construct validity, credible measurement depends on triangulation across benchmarks rather than reliance on any single test.

4 Extension: Low-Supervision Scaling & Large Language Models

Large language models (LLMs) have rapidly expanded the toolkit available for automated text analysis in the social sciences. Many prominent contemporary LLMs are transformer-based, decoder-only language models trained on massive text corpora, typically with objectives focused on next-token prediction or closely related causal language modeling tasks. These training objectives enable LLMs to execute a wide range of downstream tasks in response to natural-language prompts. Unlike the transformer encoders examined in Section 3, which return dense embedding vectors as model outputs, LLMs natively generate textual responses—such as summaries, explanations, and extracted information—in response to researcher prompts.

For our purposes, large language models can also be prompted to produce measures for text units, including document classifications and judgments of document positions on a latent dimension of interest. The use of LLMs for these kinds of measurement tasks has become increasingly common in recent work (e.g., Benoit, De Marchi, Laver, Laver, and Ma 2025; Choi, Peskoff, and Stewart 2026; Le Mens and Gallego 2025; Ornstein, Blasingame, and Truscott 2025). In practice, researchers supply an LLM with raw text and instructions specifying the measurement task, and scaled positions are obtained directly from the model's response. Indeed, LLMs can generate scaled positions for units of interest from

unfiltered text with minimal instructions—a stark contrast to the more elaborate design choices required by DFM- and embedding-based pipelines. From the researcher's perspective, this makes LLM-based measurement substantially more accessible, as the workflow does not require a high degree of technical specification at each stage of the measurement pipeline.

At first glance, this ease of use may encourage researchers to adopt a comparatively hands-off approach to measurement design, keeping both prompt specification and text filtering to a minimum, on the view that the model can infer the relevant design choices. Indeed, choices that are explicit and observable in the low-supervision pipelines we have discussed—such as how text is represented numerically and how latent positions are recovered—are folded into the LLM's response process. Once text and a prompt are supplied, researchers typically observe no intermediate stage between model input and final scaled output. Even for open-weight models whose architecture and parameters can be inspected, the computational process by which the model transforms a text and prompt into a scaled position is not readily decomposable into the kinds of interpretable measurement decisions that characterize standard low-supervision pipelines. For proprietary models—such as Claude, Gemini, and GPT—the architecture, training data, and optimization objectives are entirely undisclosed, making the measurement process even more opaque to researchers. This opacity can foster an "out of sight, out of mind" mentality, in which researchers delegate consequential measurement decisions to the model simply because the model can produce scaled outputs without those decisions being made explicit.

We argue that although LLMs have the potential to automate key design choices, this is no substitute for deliberate researcher decision-making in the measurement pipeline. Several features of LLM-based measurement make a hands-off approach difficult to justify. First, vague or underspecified prompts may fail to constrain which latent dimension the model recovers. LLMs bring to the task their own learned associations between language and concepts, acquired during training. A prompt requesting a candidate's left-right position on healthcare, for instance, may be operationalized by the model in ways that do not align with the researcher's definition of the target construct (e.g., recovering positions on reproductive healthcare instead of government intervention in healthcare). Rather than applying an explicit researcher-supplied measurement rule, the model draws on latent distinctions acquired during pretraining—distinctions whose relationship to the target construct is unknown. Second, short prompts may be especially sensitive to minor changes in wording (Barrie, Palaiologou, & Törnberg, 2024). In such cases, choices that appear benign are in fact consequential elements of measurement design, because they can alter what the model treats as relevant evidence and how it maps that evidence onto a latent position. Third, supplying long, unfiltered texts to LLMs for low-supervision scaling increases the risk that the model will apply the target measurement rule inconsistently as it processes the document. With longer inputs, the model may lose fidelity to the original task specification, leading to worse task performance (N. F. Liu et al., 2024).

Motivated by these concerns and considerations, we examine LLM-based approaches to latent measurement within the pipeline framework developed in this Element. We show that LLM-based scaling remains sensitive to the same pipeline choices examined in Section 2, and that the validation principles developed in Section 3 apply with equal force. Measures remain vulnerable to the same potential failures of construct recovery and positional accuracy. Even though LLMs can appear to collapse multiple stages of the measurement pipeline into a single prompting step, their credible use for latent scaling still requires the same disciplined attention to pipeline design and construct validation that we have advocated throughout this Element.

There are additional considerations relevant to the use of LLMs in low-supervision scaling that we do not examine directly here, but that remain consequential for the design and implementation of LLM-based measurement pipelines. Although these considerations lie beyond our immediate focus on measurement design and construct validity, they bear directly on how such approaches are deployed in practice. These include:

1. **Prompting:** An extensive literature examines prompt engineering and optimization for LLM-based applications. Our analysis centers on the contrast between minimal and maximal prompt specification, but in practice, prompt design varies along a much richer continuum, including choices about task framing, instruction sequencing, the use of exemplars (e.g., few-shot prompting), output constraints, and response formatting. For an instructional overview of practical guidance on prompt design, see Schulhoff et al. (2024) and Ornstein et al. (2025). For a discussion of prompt sensitivity, see Barrie, Palaiologou, and Törnberg (2024).

2. **Task objective:** LLM performance varies systematically across measurement tasks. In general, these models perform better when positional judgments on the target latent construct can be grounded in explicit textual evidence, and worse when the task is conceptually ambiguous, only weakly signaled in the text, or dependent on background knowledge not contained in the prompted text itself. For broader discussions of how task characteristics shape LLM task performance, see Bisbee and Spirling (2025) and Yang, Wang, Zhou, and Xu (2025).

3. **Replication:** Different LLMs can yield meaningfully different measurement outputs on the same task, and the magnitude of those differences may itself vary across applications. As we argue in Section 2.2, researchers should therefore treat model choice as a design decision and, where possible, assess the robustness of substantive conclusions across multiple LLMs. For discussions of model-specific variations and their implications for replication, see Barrie, Palmer, and Spirling (2024) and Benoit et al. (2025).

4. **Reproducibility:** A particular challenge for LLM-based measurement is that many model endpoints are not version-controlled, so the same system

may produce different outputs over time without notice. This raises concerns about reproducibility, as changes in results may reflect undocumented model updates rather than deliberate changes in the research design. Researchers should therefore document model versions whenever possible and treat non-version-controlled systems as a source of instability in the measurement pipeline (Barrie, Palmer, & Spirling, 2024).

5. **Downstream inference**: Scaled judgments produced by LLMs are subject to the same forms of prediction error that arise in other machine-learning pipelines. In LLM-based measurement, however, these errors may be further compounded by biases embedded in the model itself, spanning diverse social and political domains such as race, gender, and ideology (Lin, Wang, Guo, & Wong, 2025). This is consequential for downstream inference, because treating LLM-generated measures as error-free can both overstate the precision of estimated relationships and obscure systematic distortions in the underlying measure. For discussions of these forms of bias, see Bender, Gebru, McMillan-Major, and Shmitchell (2021). For approaches to accounting for such error in downstream inference, see Egami et al. (2024).

In the subsequent sections, we first outline our approach for producing LLM-based measures for candidates' issue-specific left-right positions. Following the pipeline framework outlined in Sections 2 and 3, we produce 24 different LLM-based measures by varying the level of text inclusion and the degree of prompt specificity—a design lever analogous to the unsupervised versus weakly supervised scaling dichotomy. We then evaluate the construct validity of these LLM-based measures using tests of predictive, convergent, and face validity, as developed in Section 3.1. Finally, drawing on the results of our construct validity analyses, we underscore several important limitations in LLM-based scaling performance. These considerations directly relate to the text inclusion and scaling choices that structure low-supervision measurement pipelines.

4.1 Pipeline Specification

The principal design choices we vary in constructing LLM-based measurement pipelines are similar to those examined in the preceding sections, spanning text inclusion and scaling. We hold text representation fixed by using a single model across all pipeline configurations: Meta's Llama 3, a 70-billion-parameter open-weight large language model trained on more than 15 trillion tokens drawn from publicly available sources (Meta AI, 2024; Touvron et al., 2023). We do not vary the choice of LLM across alternatives such as GPT, Claude, or Mistral, because the aim of this section is not to compare performance across LLMs, but to demonstrate that LLM-based measurement remains subject to the same pipeline design choices examined earlier in the Element. Our use of an open-weight model also strengthens reproducibility, since proprietary systems may change model versions or parameters without notice.

In our LLaMA-based pipelines, text inclusion and scaling decisions are communicated to the model through the prompt—a structured set of natural-language instructions that governs how the measurement task is implemented. An example prompt for the abortion measurement task is presented in the box below; all other prompts are available in Appendix Section A2.3:

```
You are scoring a U.S. congressional candidate's statement on
abortion policy.
---
Task: Score the statement's abortion-policy position on a 0-100
very left to very right scale.

DEFINITION (Abortion policy):
Statements about access to abortion, abortion bans or restrictions,
Roe v. Wade / Dobbs, federal or state abortion legislation,
exceptions (rape, incest, life of the mother), gestational limits,
abortion as a right or as murder, and Planned Parenthood ONLY when
explicitly tied to abortion.
Do NOT count contraception, reproductive education, women's
healthcare, or Planned Parenthood unless abortion is explicitly
discussed.

SCORING PRINCIPLES:

• Score ONLY the abortion-related content; ignore unrelated
  content.
• Score what the text says explicitly. Do not infer positions.
• Use the FULL range, but 0 and 100 should be RARE.
• If the text is mixed or balanced, pull toward the center (around
  50).
• If the text expresses a direction weakly or rhetorically, use
  scores near the midpoints.
• If the text discusses abortion but does not take a clear
  pro-choice or pro-life stance, score as ambiguous (near 50).

OUTPUT FORMAT:
Return ONLY valid JSON: {"Score": <number 0-100>} or {"Score":"NA"}
if the text is irrelevant to abortion policy. Do not include any
explanations.
```

For text inclusion, we vary the scope of text supplied to the model using the same three levels of aggregation examined in Section 3: paragraphs, platform points, and full campaign platforms. At each level, we supply only text from candidates that our supervised classifiers identified as discussing the target construct, consistent with the filtering procedures described in Section 3. To mitigate the risk that construct-irrelevant content in these texts influences the model's output, we further prompt the model to: (i) score only content directly related to the target issue domain, and (ii) base its assessment on input text explicitly rather than infer positions from indirect cues. This task specification builds on the prompting strategy from Le Mens and Gallego (2025).

With respect to scaling, we vary the degree of construct specification provided in the prompt, paralleling the distinction between unsupervised and weakly supervised scaling approaches outlined in Section 2.3. In the unsupervised pipelines, we provide minimal guidance: defining the target construct, outlining basic scoring principles, and specifying the output format. In the weakly supervised pipelines, we supplement this prompt with additional calibration guidance. This added prompting provides the model with more explicit information about the types of positions that should map onto different regions of the scale. An example of this prompt extension for the abortion measurement task is as follows:

```
You are scoring a U.S. congressional candidate's statement on
abortion policy.
```

[...]

Scoring Calibration:

0–15 Very left: Abortion access without restrictions; abortion framed as a fundamental right; calls to codify Roe v. Wade; repeal Hyde Amendment or global gag rule; "abortion is healthcare," "reproductive justice," "reproductive rights."

20–35 Left: General pro-choice rhetoric; protecting women's right to choose; abortion should be available; may express personal opposition but supports access; no explicit legislative detail.

40–60 Centrist: Abortion available with restrictions (e.g., trimester or 20-week limits); balanced mix of pro-choice and pro-life language; emphasis on reducing abortions without banning them; discussion of adoption or education without a clear policy stance.

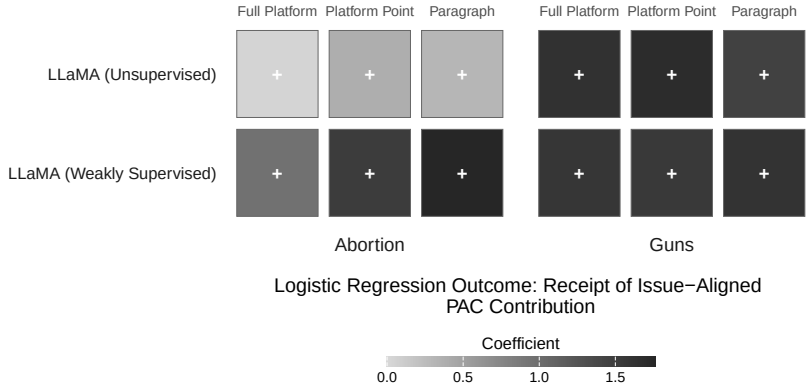
65–80 Right: Pro-life rhetoric; abortion unlawful in most cases but with common exceptions (rape, incest, child pregnancy, life of the mother); "right to life," "life is precious."

85–100 Very right: No abortions under any circumstances except possibly life of the mother; abortion framed as murder or homicide; "life begins at conception"; calls for federal bans or outlawing abortion nationwide.

[...]

Taken together, these design choices yield 24 LLM-based pipeline configurations: 3 text-inclusion rules × (1 LLM) × 2 scaling approaches, for 6 configurations per issue.

Figure 15 Predictive Validity: PAC Contributions and Candidate Issue Extremity, with LLM-Based Pipelines



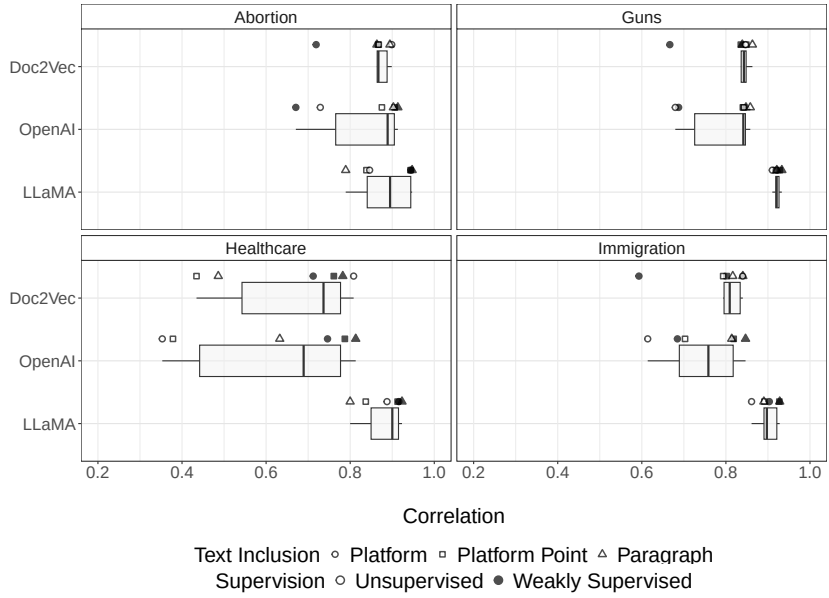
4.2 Validating LLM-Based Pipelines

We evaluate the construct validity of our LLaMA-based pipeline configurations by first assessing predictive validity. Figure 15 reports, for each pipeline configuration, the estimated relationship between candidate issue-position extremity and the likelihood of receiving a contribution from an issue-aligned PAC. This predictive-validity test follows the same specification as the one reported in Figure 8 in Section 3.2. A pipeline that recovers a construct-valid measure should yield a positive, statistically significant association between issue-specific extremity and the probability of receiving a positionally aligned PAC contribution. Grayscale shading indicates the magnitude of the estimated coefficient, and the overlaid symbols denote the direction of the relationship. Full model outputs are available in Appendix Section B2.

The logistic regression coefficients reported in Figure 15 demonstrate that LLaMA-based pipelines consistently satisfy the predictive validity benchmark across both the abortion and gun policy domains. All estimated coefficients are positive and statistically significant, indicating that candidates expressing more extreme issue positions are more likely to receive contributions from positionally aligned PACs. This pattern holds across both supervision levels and all three text inclusion regimes, suggesting that LLaMA-based scaling recovers sufficient construct-relevant signal to distinguish candidates whose issue-specific positioning attracts issue-aligned PACs.

We next assess convergent validity by correlating LLaMA-scaled positions with average human-coded ratings of candidates' issue-specific left-right positioning, following the same validation strategy used in Figure 9 in Section 3.2. Each panel corresponds to an issue domain, with shape indicating the level of text inclusion and fill distinguishing unsupervised from weakly supervised scaling. Boxplots summarize the distribution of correlations across pipeline

Figure 16 Convergent Validity: Human Text Ratings & Scaled Position Measures, with LLM-Based Pipelines

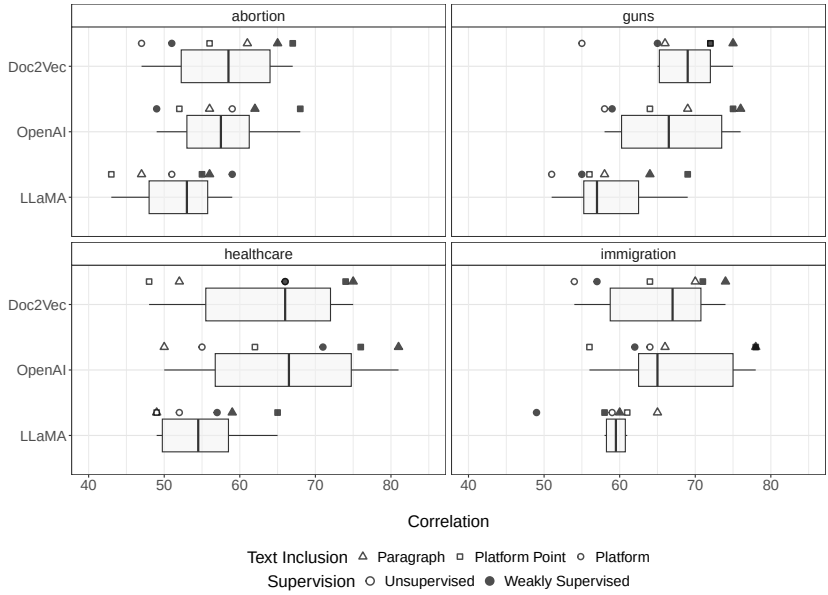


configurations, with the median shown by the vertical line inside each box. Higher correlations indicate stronger alignment between pipeline outputs and human judgments of the same underlying text. We include correlations from the best-performing pipelines in Section 3 (Doc2Vec and OpenAI, Paragraph, Weakly Supervised) as reference points to benchmark the relative performance of LLaMA-based pipelines.

Figure 16 shows that LLaMA-based pipelines achieve consistently high convergent validity across all four issue domains, with correlations that are, on average, higher than those produced by the best-performing Doc2Vec and OpenAI pipelines. For abortion and guns—issue domains in which the aforementioned pipelines performed particularly well—LLaMA-based configurations match or exceed the strongest observed pipeline performance. The contrast in correlational performance is most striking for healthcare and immigration, where the best-performing LLaMA pipelines achieve correlations of 0.92 and 0.93, respectively, compared to 0.81 and 0.84 for the strongest DocVec configurations and 0.81 and 0.85 for the strongest OpenAI configurations.

LLaMA-based pipelines also exhibit substantially narrower boxplots than Doc2Vec and OpenAI across all four issue domains, indicating that convergent validity in these pipelines is less sensitive to text inclusion and supervision decisions. This might seem to vindicate the hands-off approach described at the outset of this section—if LLM-based scaling is robust to pipeline decisions

Figure 17 Face Validity: Proportion of Construct-Relevant Keyness Terms, with LLM-Based Pipelines



requiring less supervision and text filtering, perhaps the model can indeed infer the relevant design choices under minimal researcher guidance. But this apparent robustness on convergent validity does not necessarily imply stronger construct validity for LLaMA-based pipelines overall.

Figure 17 plots the proportion of high-keyness terms that are substantively aligned with the target construct. Following the same procedure used to produce Figure 10 in Section 3.4, we identify the top 50 keyness terms associated with texts positioned at the left and right extremes of each issue dimension, relative to more moderately positioned texts. We then assess whether these terms directly engage the target issue domain or instead reflect general political rhetoric, unrelated policy areas, or other construct-irrelevant content. The figure follows the same visual conventions as Figure 16. Higher proportions indicate stronger face validity—that extreme texts are distinguished from moderate texts on the basis of construct-relevant language. We include Doc2Vec and OpenAI results alongside the LLaMA-based pipelines to provide a point of reference for evaluating LLaMA's face validity performance.

The face validity results in Figure 17 reveal a striking tension with the convergent-validity evidence. Despite achieving the highest correlations with human ratings across all four issue domains, LLaMA-based pipelines consistently produce lower proportions of construct-relevant keyness terms relative to Doc2Vec and OpenAI pipelines. This pattern holds across all four domains, both

at the median and at the upper end of each distribution: even the strongest LLaMA configurations fall short of most Doc2Vec and OpenAI configurations. The gap is largest in healthcare and immigration—the two domains in which LLaMA's convergent-validity advantage was greatest—where LLaMA distributions cluster in the mid-50s to around 60%, while the upper ends of the Doc2Vec and OpenAI distributions generally extend above 70%. These results indicate that the terms most responsible for distinguishing extreme from moderate texts in LLaMA-based pipelines are less directly tied to the target latent dimension. Moreover, the LLaMA boxplots in Figure 17 are substantially more dispersed than those in Figure 16, with the weakest face validity concentrated among pipelines using broader text inclusion and less supervision—a sensitivity to pipeline design choices that the convergent-validity results did not reveal.

How should we square this tension? The predictive and convergent validity evidence suggests that LLaMA-based pipelines perform well—and consistently—regardless of text inclusion or supervision decisions, lending apparent support to a hands-off approach in which the model resolves key design choices internally. But the face validity results tell a different story. The textual features discriminating scaled positions in LLaMA pipelines are substantially less construct-related than those produced by the best-performing Doc2Vec and OpenAI pipelines, and this disparity is most pronounced among LLaMA pipelines that rely on the least supervision and the broadest text inclusion. This divergence across validation exercises reinforces a point we develop in Section 3.1: no single validity test is dispositive, because each is sensitive to different forms of measurement failure. A measure that performs well on one benchmark may fail on another, and only their combined evidence provides a credible basis for determining construct recovery. To better understand when and how LLM-based scaling falls short, we next examine two specific failure modes suggested by these results.

4.3 Measurement Failure in LLM-Based Pipelines

The results above suggest that the apparent robustness of LLM-based measurement to certain validation tasks masks important forms of measurement failure. We next examine two specific mechanisms underlying these failures. The first concerns text inclusion: we demonstrate that as text inputs grow longer, the model's fidelity to the prompted measurement task deteriorates. The second concerns positional discrimination: we show that LLMs generally collapse what should be a one-dimensional continuum into a small number of de facto categories, and that this tendency is exacerbated by a lack of prompt specificity.

4.3.1 Pitfalls of long-text inclusion

The convergent-validity results in Figure 16 suggest that text-inclusion decisions have only a limited impact on the performance of LLaMA-based

measurement pipelines. A closer examination of the model's outputs, however, shows that this apparent robustness erodes as text inputs grow longer, making the model less likely to return valid scaled positions.

In constructing the LLaMA-based pipelines, we prompt the model to return a scaled position in the format `"Score":<number 0-100>`. When the supplied text does not pertain to the target issue, the model is instead instructed to return `"Score":"NA"`. In practice, however, the model sometimes returned `"NA"` or an empty score field even for text inputs that our supervised classifier had already identified as construct-relevant.[29] Each such nonresponse admits two interpretations: either the model correctly identified a text that our supervised classifier had mistakenly labeled as relevant (a classifier false positive), or it failed to recognize construct-relevant content that was genuinely present in the text (an LLM false negative).

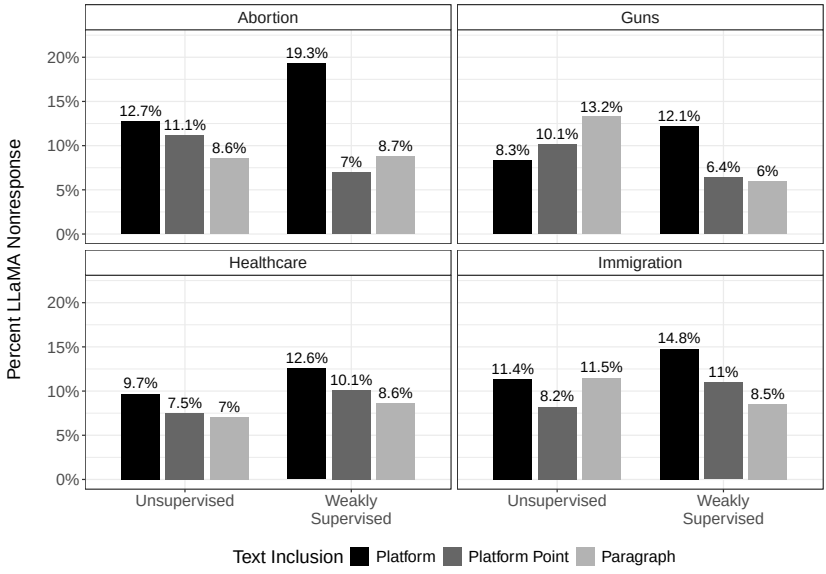
Figure 18 reports the rate of LLM nonresponse across pipeline configurations. The central pattern is clear: full-platform text inclusion produces substantially higher nonresponse rates than platform-point or paragraph inclusion across issue domains and supervision levels. The disparity is especially pronounced for abortion under weak supervision, where nearly 20% of full-platform inputs—641 of 3,272 prompted inputs—failed to return a valid score.

To determine whether these nonresponses reflect a genuine absence of construct-relevant content or instead failures by the LLM to identify relevant text, we conduct a manual audit of `"NA"` responses at two levels of text inclusion. For the abortion task, we randomly sampled 100 paragraph-level texts that received `"NA"` scores and read each to determine whether it contained construct-relevant content. Of these, 56 contained no abortion-related material, indicating that the model had correctly identified a false positive from our supervised classifier. The remaining 44 contained content that directly engaged the target dimension, which the LLM failed to recognize. This result suggests that nonresponse in paragraph-level pipelines reflects a mix of valid filtering and relevant content missed by the LLM.

We repeated this exercise for full-platform texts that received `"NA"` scores. The results were markedly weaker: only 29 of 100 `"NA"` responses reflected a genuine absence of construct-relevant content. This suggests that, under full-platform inclusion, nonresponse is far more likely to indicate LLM measurement failure than to reflect successful nonresponse to non-construct-relevant text. This deterioration is consistent with documented limitations in the long-context processing of large language models. Research on long-context fidelity has shown that models can progressively lose adherence to the original task specification as document length increases (Levy, Jacoby, & Goldberg, 2024; Wu et al., 2025). A related body of work on positional attention shows that content located in the middle of long documents tends to receive less weight in task

[29] Occasionally, the model produced nonsensical text rather than `"NA"` or an empty field. Failures to return valid annotations have been documented in prior work (e.g., Yang et al. 2025). In our data, this failure mode was exceptionally rare: for instance, in the abortion domain, it occurred in 10 of 3,270 prompt inputs.

Figure 18 Proportion of Input Prompts for Scaled Positions Generating Non-Response Outputs



performance than content appearing near the beginning or end (Hosseini, Castro, Ghinassi, & Purver, 2025). Both mechanisms help explain why full-platform inclusion—which embeds construct-relevant signal within large volumes of unrelated policy content—produces substantially higher rates of measurement failure than paragraph-level inclusion.

The preceding analysis focuses on false negatives, in which the LLM fails to return a scaled position for units with construct-relevant text. The complementary risk is false positives, in which the LLM assigns a scaled position to units that do not, in fact, engage the target construct. Because our existing pipelines only prompt the LLM to return scaled positions for candidates preclassified as discussing a targeted issue domain, we cannot directly evaluate this failure mode with the available estimates. Therefore, to evaluate false positives directly, we supplied the model with all full-platform texts that our classifier identified as *not* discussing abortion. Of 2,627 such texts, the model nonetheless returned a scaled position for 152. We randomly sampled 100 of these and found that, in 76 cases, the text contained no abortion-related content; the model inferred a position from general political cues in the text despite explicit prompt instructions to score only construct-relevant material.[30] This result shows that prompt-based instructions alone are insufficient to ensure that LLMs restrict

[30]Common triggers for these false positives included partisan language, references to women's health in non-abortion contexts, and general statements about constitutional rights—cues that are politically informative but do not engage the target dimension.

positional scaling to construct-relevant text, especially when documents are lengthy and contain substantial irrelevant content.

Beyond its consequences for measurement quality, broad text inclusion also imposes substantial computational costs. In our application, processing the full corpus of campaign platforms through LLaMA cost $6.27 and required 12.3 million input tokens. Restricting the input to construct-relevant paragraphs would reduce this cost by a factor of 26.4 for the abortion measurement task and 10.2 for the healthcare measurement task. In practice, these cost differences compound quickly, since researchers typically repeat the measurement process across multiple prompt specifications, estimate models under alternative configurations for robustness checks, and sometimes reprompt texts when inputs require correction. Our corpus is also modest relative to many text-based measurement applications in the social sciences, so for larger corpora, the cost differential between filtered and unfiltered inputs would scale proportionally.

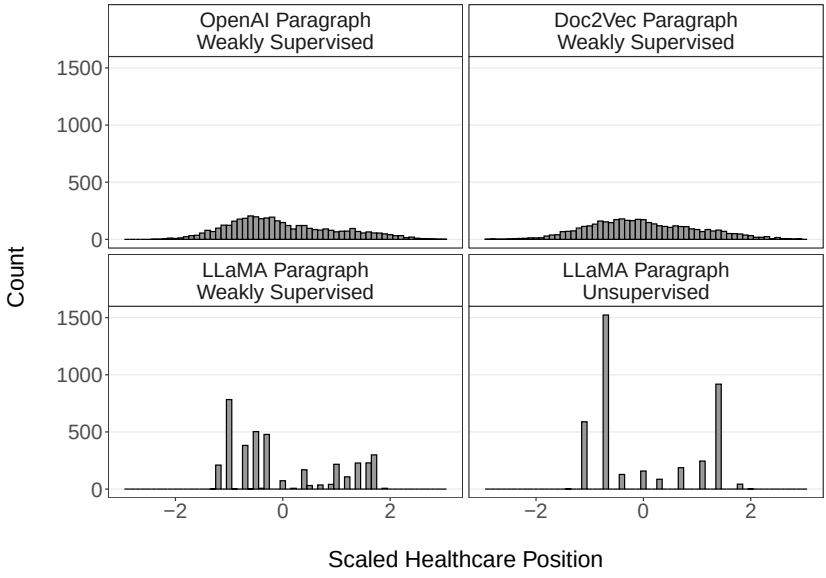
Taken together, these results show that text inclusion is not a neutral decision in LLM-based scaling. Longer documents increase the risk that the model will fail to recognize construct-relevant content and generate positions from construct-irrelevant cues. These unfiltered texts also substantially increase the computational cost of LLM-based pipelines. Both drawbacks—lower measurement fidelity and higher computational expense—are largely masked by aggregate validity metrics and become apparent only through direct inspection of model outputs and associated texts.

4.3.2 Positional discrimination & clustering

The face-validity results in Figure 17 show that LLaMA-based pipelines consistently produce lower proportions of construct-relevant keyness terms than Doc2Vec and OpenAI pipelines, with the share of construct-relevant terms often at or below 60%. In Section 3.4, we treated proportions below this threshold as evidence of weak construct discrimination across scaled positions. These face-validity results are difficult to reconcile with LLaMA's strong performance on the predictive and convergent-validity benchmarks in Figures 15 and 16. If the model recovers the target construct well enough to align closely with human ratings and to predict PAC contributions, why do face validity indicators suggest only weak construct recovery?

One possible explanation is that the model is not producing well-differentiated continuous positions along the target dimension. If scaled positions cluster at a small number of values rather than spreading across a continuum, the distinction between extreme and moderate texts becomes less substantively meaningful, leaving the keyness procedure with limited positional variation from which to recover construct-specific differences. Figure 19 investigates this possibility by plotting the distribution of scaled healthcare positions for LLaMA-based pipelines, as well as OpenAI and Doc2Vec (Paragraph, Weakly Supervised) to

Figure 19 Distribution of Scaled Healthcare Positions across Pipeline Configurations



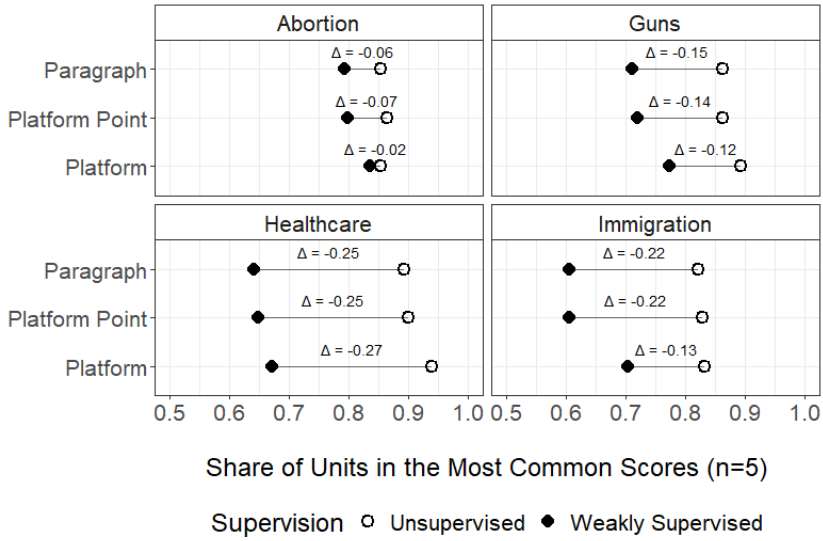
provide a point of reference.[31] Histograms have a bin width of 0.1.

The histogram distributions in Figure 19 illustrate that the OpenAI and Doc2Vec pipelines produce smooth, continuous distributions spread across the full range of the scale. The LLaMA pipelines, by contrast, exhibit severe score clustering—what recent work terms heaping (Licht et al., 2025)—in which a large share of scaled positions concentrate at a small number of discrete values. Such clustering is especially pronounced in the unsupervised LLaMA configuration in which 1,523 units, or 39% of all scaled healthcare observations, receive a score of −0.73. Given that healthcare is the most substantively nuanced of the four issue domains we examine, it is implausible that so many candidate–issue–year observations occupy an identical position on the target dimension. Importantly, this clustering is not confined to a single issue domain or to paragraph-level text inclusion; it also appears across the other issue areas and under both platform-point and full-platform pipeline configurations.

This distributional pattern helps to explain the poor face validity performance of LLaMA pipelines. Recall that the keyness diagnostic compares the language of texts assigned to the most extreme positions with that of texts assigned to more moderate positions. When scaled positions cluster around a handful of values, the boundary between extreme and moderate positioning in measurement is poorly defined, and the terms that distinguish these groups are unlikely to

[31] Recall that all measures are rescaled to have mean zero and unit variance, making the resulting distributions directly comparable.

Figure 20 Scaled Position Concentration by Supervision Level and Text Inclusion



reflect substantively meaningful positional differences. In short, there is little textual discrimination for the keyness procedure to recover because there is little positional discrimination in the scaled positions themselves.

The strong convergent-validity results for LLaMA-based pipelines in Figure 16 should be understood accordingly. High correlations with human ratings likely reflect the model's ability to recover broad ordinal distinctions that align well with the five-point rating scale employed by human coders (see Appendix Section B1 for greater detail). However, agreement with a coarse benchmark does not imply that the resulting measure is continuous in any meaningful sense. The clustered LLaMA distributions in Figure 19 instead suggest that the model is preserving broad ordering while collapsing many observations into a small number of de facto categories, rather than producing the finer positional discrimination that continuous scaling is intended to capture.

Figure 20 shows that the clustering problem becomes systematically more severe as prompt supervision is reduced. For the LLaMA pipeline configuration, we compute the share of candidate–issue–year units assigned to the five most common scaled position values and compare that share across weakly supervised and unsupervised prompting. The figure shows the difference in this concentration (Δ) between weakly supervised and unsupervised prompts. Negative values indicate that weakly supervised pipelines distribute scaled positions more evenly across the scale.

Every comparison in Figure 20 presents negative changes in concentration, demonstrating that unsupervised prompting consistently produces greater score

concentration than weakly supervised prompting. The problem is most severe for full-platform pipeline configurations in the healthcare and guns domains, where unsupervised prompting compresses 90–95% of all scaled positions into just five values. These results indicate that low-specificity prompting is ill-suited to continuous measurement tasks, because it leaves the model underspecified with respect to the distinctions that should separate nearby positions on the latent scale. Even when aggregate correlational metrics appear strong—as in Figure 16, where unsupervised LLaMA pipelines rival or exceed Doc2Vec and OpenAI—such benchmarks can mask a more fundamental failure of positional discrimination. These results have clear implications for prompt design. Prompt specifications should define not only the endpoints of the target continuum, but also the kinds of incremental differences that ought to distinguish nearby positions along the scale. Rather than taking a hands-off approach, researchers should specify this granularity directly rather than relying on the model to infer the relevant distinctions.

Pairwise comparison frameworks, in which the model evaluates the relative positions of two texts rather than assigning each a fixed score, offer a promising way to mitigate the clustering problem (Licht et al., 2025). By shifting the task from absolute placement to relative judgment, such designs may better align with the comparative strengths of large language models and reduce the tendency to collapse a continuous latent scale into a small number of de facto categories. That said, pairwise approaches also impose substantial computational costs, since the number of required comparisons grows with corpus size and becomes especially burdensome when the texts themselves are long. Their practical appeal, therefore, depends on a tradeoff between improved positional discrimination and the added expense of implementation at scale.

4.4 Summary of Practical Guidance

The results in this section demonstrate that LLM-based measurement pipelines do not eliminate the design choices that characterize low-supervision scaling. Effective LLM-based scaling still requires selecting a model that is ideally version-controlled to ensure reproducibility, filtering input texts to construct-relevant content where possible, and crafting prompts with sufficient specificity to guide the model's positional judgments. The measurement pipeline, in short, does not go away; it is merely reconfigured.

Moreover, the ostensibly accessible implementation of LLM-based measurement pipelines is less straightforward than it initially appears. Many of the failure modes we identify—nonresponse driven by long-context degradation, false positives generated from construct-irrelevant cues, and severe score clustering under low-specificity prompting—are obscured by aggregate validity metrics and become visible only through direct inspection of model outputs. This places a greater burden on the researcher to diagnose measurement failures and, where possible, to correct them. Prompt engineering is central to this process and often

requires running the model with multiple prompt specifications, each of which is inspected and validated. Critically, each iteration in which the prompt is revised should be validated against independent evidence—for example, a different set of hand-labeled observations rather than the one used in earlier rounds of prompt development; otherwise, prompt refinements risk overfitting to idiosyncratic features of the original validation sample rather than improving measurement of the target construct. This, in turn, compounds the computational, financial, and validation burden of LLM-based measurement rather than eliminating it.

These considerations reinforce the broader argument of this paper. The divergence between LLaMA's strong convergent validity and weak face validity illustrates precisely why construct validation demands multiple, complementary diagnostics rather than reliance on any single benchmark. Apparent robustness on one dimension of validity can mask substantive measurement failures that only surface under alternative forms of evaluation—making thorough, multi-faceted validation all the more essential when the pipeline itself offers fewer points of direct researcher oversight.

5 Conclusion

This Element has outlined a unified framework for constructing and validating latent measures from text using low-supervision scaling approaches. We have argued that credible text-based measurement requires researchers to navigate a sequence of interconnected design choices in the text-to-measure pipeline—which texts to include, how to represent them numerically, and how to scale those representations onto a target latent dimension—and that each of these choices carries consequences for construct validity that can only be assessed through systematic, multi-faceted validation.

Our systematic comparison of more than one hundred pipeline configurations demonstrates that design choices at each pipeline stage shape downstream measures in ways that are often non-obvious, with consequences that depend on choices made elsewhere in the pipeline. Text inclusion decisions determine the mix of construct-relevant and non-relevant content available for scaling, and our results show that filtering texts to the most construct-relevant units more consistently places observations at their appropriate positions on the target latent dimension. Text representation determines which aspects of language—term frequency, distributional semantics, compositional meaning—are preserved for scaling, and matching representation to the linguistic form of the construct-relevant signal proved critical for measurement quality. Scaling procedures, whether unsupervised or weakly supervised, can in some configurations partially compensate for weaker text inclusion or representation decisions, but they do not substitute for strong pipeline design across all stages.

Our examination of large language models within this pipeline framework reinforces these conclusions. LLM-based pipelines do not bypass the design choices that define the text-to-measure pipeline. The apparent robustness

of LLM-based scaling on some validity benchmarks masks important forms of measurement failure that emerge only under closer scrutiny and across multiple validation benchmarks. These findings underscore a central lesson of this Element: no single validity test is dispositive, and only the combined evidence from multiple complementary assessments provides a credible basis for evaluating construct recovery.

Researchers who construct latent measures from text assume a distinctive set of responsibilities. They must articulate a clear definition of the target construct, select text data that supports valid measurement of that construct, and make principled decisions at each stage of the pipeline. They must validate the resulting measures against multiple complementary benchmarks, recognizing that apparent success on any single diagnostic may conceal substantive measurement failures. And they must document their design choices in sufficient detail to permit replication and critical evaluation by others. These responsibilities are not burdens imposed by methodological convention; they are the conditions under which text-based measures can credibly serve as inputs to empirical inquiry. The growing accessibility of low-supervision scaling methods makes attending to these responsibilities all the more important, as the ease of producing measures must not outpace the rigor with which those measures are evaluated.

The approaches to low-supervision scaling examined in this Element span multiple decades of methodological development in computational text analysis. From bag-of-words scaling models such as Wordfish and Wordscores (Laver et al., 2003; Slapin & Proksch, 2008), which recover latent positions from cross-document variation in term frequencies, to embedding-based methods that exploit the learned geometry of semantic space, to large language models that leverage vast pretrained representations to assign scaled positions via natural-language prompts. The toolkit for low-supervision scaling has expanded dramatically in both scope and sophistication. Yet the core measurement challenges persist across all of these approaches. No method, however technically advanced, eliminates the need for researchers to clearly define what they intend to measure, to justify the design choices that produce those measures, and to assess whether the resulting quantities capture the construct they claim to represent.

As the toolkit for low-supervision scaling continues to expand—through new model architectures, prompting strategies, embedding techniques, and approaches to dimension recovery—the framework we outline here will remain applicable. The text-to-measure pipeline is not specific to any particular generation of methods; rather, it reflects the underlying logic of transforming qualitative political text into quantitative measures. Any such approach must, whether implicitly or explicitly, make choices about what text enters the analysis, how that text is represented numerically, and how those representations are converted into scaled positions. Our framework makes those choices explicit, organizes them into analytically distinct stages, and creates a common basis for evaluating their consequences. The specific methods that populate each stage of the pipeline will continue to evolve, but the need to justify design choices, assess their measurement implications, and report them transparently will remain.

References

Adcock, R., & Collier, D. (2001). Measurement validity: A shared standard for qualitative and quantitative research. *American political science review*, *95*(3), 529–546.

Aldrich, J. H., & McKelvey, R. D. (1977). A method of scaling with applications to the 1968 and 1972 presidential elections. *American Political Science Review*, *71*(1), 111–130.

American Political Science Association, Committee on Political Parties. (1950). *Toward a more responsible two-party system*. New York: Rinehart.

Ansolabehere, S., Snyder, J. M., & Stewart, C. (2001). Candidate positioning in u.s. house elections. *American Journal of Political Science*, *45*(1).

Bailey, M., & Reese, B. (2025, January). *Ideological moderation and success in u.s. elections, 2020–2022*. Retrieved from `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5169898` (32 pages. Posted March 21, 2025)

Baker, A. E. (2016). Do interest group endorsements cue individual contributions to house candidates? *American Politics Research*, *44*(2).

Barbera, P. (2015). Birds of the same feather tweet together: Bayesian ideal point estimation using twitter data. *American Political Science Review*, *23*(1), 76-91.

Barrie, C., Palaiologou, E., & Törnberg, P. (2024). Prompt stability scoring for text annotation with large language models. *arXiv preprint arXiv:2407.02039*.

Barrie, C., Palmer, A., & Spirling, A. (2024). Replication for language models problems, principles, and best practice for political science. *URL: https://arthurspirling.org/documents/BarriePalmerSpirling TrustMeBro.pdf*.

Barrie, C., Palmer, A., & Spirling, A. (2025). *Replication for language models: Problems, principles, and best practices for political science* (Tech. Rep.). Working paper, version November 4, 2025. Retrieved from `https://arthurspirling.org/documents/BarriePalmerSpirling_TrustMeBro.pdf`

Bellodi, L. (2023). A dynamic measure of bureaucratic reputation: New data for new theory. *American Journal of Political Science*, *67*(4), 880–897.

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 acm conference on fairness, accountability, and transparency* (pp. 610–623).

Benoit, K., De Marchi, S., Laver, C., Laver, M., & Ma, J. (2025). Using large language models to analyze political texts through natural language understanding. *American Journal of Political Science*.

Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., & Matsuo, A. (2018). quanteda: An r package for the quantitative analysis of textual data. *Journal of Open Source Software*, *3*(30), 774–774.

Bisbee, J., & Spirling, A. (2025). What to do when humans are no longer the gold standard large language models, state of the art and robustness for politics research. *URL: https://github.com/ArthurSpirling/futureProofR/blob/main/Bisbee$_S$pirling$_h$uman$_g$old*

Bonica, A. (2014). Mapping the ideological marketplace. *American Journal of Political Science*, *58*(2), 367-386.

Bonica, A., & Li, Z. (2021). *Inferring candidates' issue-specific positions from itemized campaign contributions using supervised machine learning.* `https://www.dropbox.com/scl/fi/ibr2a47jb4e04kf31h4hq/Bonica-and-Li-2021.pdf`. (Presented at the 2018 Annual Meeting of the Midwest Political Science Association)

Boutyline, A., & Johnston, E. (2025). *Forging better axes: Evaluating and improving the reliability of semantic dimensions in word embeddings.* SocArXiv.

Broockman, D. E. (2016). Approaches to studying policy representation. *Legislative Studies Quarterly*, *41*(1), 181-215.

Carlson, D., & Montgomery, J. M. (2017). A pairwise comparison framework for fast, flexible, and reliable human coding of political texts. *American Political Science Review*, *111*(4), 835–843.

Carmines, E. G., & Zeller, R. A. (1979). *Reliability and validity assessment*. Beverly Hills, CA: Sage Publications.

Case, C. R. (2027). Measuring strategic positioning in congressional elections. *The Journal of Politics*. (Forthcoming) doi: 10.1086/738502

Case, C. R., & Porter, R. (2026, January). *Strategic heterogeneity in issue positioning: Evidence from congressional campaigns.* Retrieved from `https://osf.io/preprints/osf/zhrmv_v5` (OSF Preprint)

Choi, D., Peskoff, D., & Stewart, B. M. (2026). Fine-tuned large language models can replicate expert coding better than trained coders: a study on informative signals sent by interest groups. *Political Science Research and Methods*, 1–19. doi: 10.1017/psrm.2025.10086

Colao, J., Broockman, D. E., Huber, G. A., & Kalla, J. L. (2025). *Tracing polarization's roots: A panel study of voter choice in congressional primary and general elections.*

Converse, P. E. (1964). The nature of belief systems in mass publics. *Critical Review*(1-74).

Däubler, T., Benoit, K., Mikhaylov, S., & Laver, M. (2012). Natural sentences as valid units for coded political texts. *British Journal of Political Science*, *42*(4), 937–951.

Denny, M. J., & Spirling, A. (2018). Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political analysis*, *26*(2), 168–189.

Downs, A. (1957). An economic theory of political action in a democracy. *Journal of Political Economy*, *65*, 135-150.

Druckman, J. N., Kifer, M. J., & Parkin, M. (2009). Campaign communications in us congressional elections. *American Political Science Review*, *103*(3),

343–366.

Duan, Z., Shao, A., Hu, Y., Lee, H., Liao, X., Suh, Y. J., ... Yang, S. (2025). Constructing vec-tionaries to extract message features from texts: A case study of moral content. *Political Analysis*, 1–21.

Egami, N., Hinck, M., Stewart, B. M., & Wei, H. (2024). *Using large language model annotations for the social sciences: A general framework of using predicted variables in downstream analyses.* Retrieved from `https://naokiegami.com/paper/dsl_ss.pdf` (Working paper, version November 17, 2024)

Ettinger, A. (2020). What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, *8*, 34–48.

Fenno, R. F. (1978). *Home style: House members in their districts*. New York: Longman.

Finke, D., & Risse, T. (2024). Multidimensional conflicts over disarmament and international security: analyzing speeches in the first committee of the un general assembly. *Political Science Research and Methods*, 1–18.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930–1955. In *Studies in linguistic analysis* (pp. 1–32). Oxford: Basil Blackwell. (Special volume of the Philological Society)

Fouirnaies, A., & Hall, A. B. (2014). The financial incumbency advantage: Causes and consequences. *Journal of Politics*, *76*(3), 711-724.

Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, *115*(16), E3635–E3644.

Grand, G., Blank, I. A., Pereira, F., & Fedorenko, E. (2022). Semantic projection recovers rich human knowledge of multiple object features from word embeddings. *Nature human behaviour*, *6*(7), 975–987.

Grimmer, J., Roberts, M. E., & Stewart, B. M. (2022). *Text as data: A new framework for machine learning and the social sciences*. Princeton, NJ: Princeton University Press.

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

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., & Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 8342–8360).

Harris, Z. S. (1954). Distributional structure. *Word*, *10*(2-3), 146–162. doi: 10.1080/00437956.1954.11659520

Hetherington, M. J. (2001). Resurgent mass partisanship: The role of elite polarization. *American Journal of Political Science*, *93*(619-631).

Hosseini, P., Castro, I., Ghinassi, I., & Purver, M. (2025). Efficient solutions for an intriguing failure of llms: Long context window does not mean llms can analyze long sequences flawlessly. In *Proceedings of the 31st*

international conference on computational linguistics (pp. 1880–1891).
Jackman, S. (2008). Measurement. In J. M. Box-Steffensmeier, H. E. Brady, & D. Collier (Eds.), *The oxford handbook of political methodology* (pp. 119–151). Oxford: Oxford University Press.
Kennedy, R., Clifford, S., Burleigh, T., Waggoner, P. D., Jewell, R., & Winter, N. J. (2020). The shape of and solutions to the mturk quality crisis. *Political Science Research and Methods*, *8*(4), 614–629.
Kim, T. (2023). Violent political rhetoric on twitter. *Political science research and methods*, *11*(4), 673–695.
King, G., Keohane, R. O., & Verba, S. (2021). *Designing social inquiry: Scientific inference in qualitative research*. Princeton university press.
King, G., Lam, P., & Roberts, M. E. (2017). Computer-assisted keyword and document set discovery from unstructured text. *American Journal of Political Science*, *61*(4), 971–988.
Kitagawa, R., & Shen-Bayh, F. (2024). Measuring political narratives in african news media: A word embeddings approach. *The Journal of Politics*, *86*(3), 1087–1092.
Kozlowski, A. C., Taddy, M., & Evans, J. A. (2019). The geometry of culture: Analyzing the meanings of class through word embeddings. *American Sociological Review*, *84*(5), 905–949.
Lau, J. H., & Baldwin, T. (2016). An empirical evaluation of doc2vec with practical insights into document embedding generation. In *Proceedings of the 1st workshop on representation learning for nlp* (pp. 78–86).
Lauderdale, B. E., & Herzog, A. (2016). Measuring political positions from legislative speech. *Political Analysis*, *24*(3), 374–394.
Laver, M., Benoit, K., & Garry, J. (2003). Extracting policy positions from political texts using words as data. *American political science review*, *97*(2), 311–331.
Layman, G. C., & Carsey, T. M. (2002). Party polarization and" conflict extension" in the american electorate. *American Journal of Political Science*, 786–802.
Layman, G. C., Carsey, T. M., & Horowitz, J. M. (2006). Party polarization in american politics: Characteristics, causes, and consequences. *Annu. Rev. Polit. Sci.*, *9*(1), 83–110.
Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *International conference on machine learning* (pp. 1188–1196).
Le Mens, G., & Gallego, A. (2025). Positioning political texts with large language models by asking and averaging. *Political Analysis*, *33*(3), 274–282.
Levy, M., Jacoby, A., & Goldberg, Y. (2024). Same task, more tokens: the impact of input length on the reasoning performance of large language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 15339–15353).
Li, Z. (2018). How internal constraints shape interest group activities: Evidence

from access-seeking pacs. *American Political Science Review*, *112*(4), 792-808.

Licht, H., Sarkar, R., Wu, P. Y., Goel, P., Stoehr, N., Ash, E., & Hoyle, A. M. (2025). Measuring scalar constructs in social science with llms. In *Proceedings of the 2025 conference on empirical methods in natural language processing* (pp. 32132–32159).

Lin, L., Wang, L., Guo, J., & Wong, K.-F. (2025). Investigating bias in llm-based bias detection: Disparities between llms and human perception. In *Proceedings of the 31st international conference on computational linguistics* (pp. 10634–10649).

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, *12*, 157–173.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., . . . Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lowe, W. (2008). Understanding wordscores. *Political Analysis*, *16*(4), 356–371.

Lowe, W., & Benoit, K. (2013). Validating estimates of latent traits from textual data using human judgment as a benchmark. *British Journal of Political Science*, *21*, 298–313.

Marble, W., & Tyler, M. (2022). The structure of political choices: Distinguishing between constraint and multidimensionality. *American Journal of Political Science*, *30*, 328–345.

Martin, L. W., & Vanberg, G. (2008). A robust transformation procedure for interpreting political text. *Political Analysis*, *16*(1), 93–100.

Meta AI. (2024, April). *Introducing meta llama 3: The most capable openly available llm to date.* Retrieved 2026-03-28, from `https://ai.meta.com/blog/meta-llama-3/`

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, *26*.

Motolinia, L. (2025). When reelection increases party unity: Evidence from parties in mexico. *British Journal of Political Science*, *55*, e55.

Munger, K. (2021). Don't@ me: Experimentally reducing partisan incivility on twitter. *Journal of Experimental Political Science*, *8*(2), 102–116.

Napolio, N. G. (2025). Measuring executive agency ideology using large language models. *Journal of Political Institutions and Political Economy*, *6*(2), 161–180.

Nussbaum, Z., Morris, J. X., Duderstadt, B., & Mulyar, A. (2024). Nomic embed: Training a reproducible long context text embedder. *arXiv preprint arXiv:2402.01613*.

Ornstein, J. T., Blasingame, E. N., & Truscott, J. S. (2025). How to train your stochastic parrot: Large language models for political texts. *Political*

Science Research and Methods, *13*(2), 264–281.
Park, J. Y., & Montgomery, J. M. (2025). Toward a framework for creating trustworthy measures with supervised machine learning for text. *Political Science Research and Methods*, 1–17.
Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).
Poole, K. T., & Rosenthal, H. (1985). A spatial model for legislative roll call analysis. *American Journal of Political Science*, *29*(2), 357-384.
Porter, R., Case, C. R., & Treul, S. A. (2025). Campaignview, a database of policy platforms and biographical narratives for congressional candidates. *Scientific Data*, *1237*.
Porter, R., Treul, S. A., & McDonald, M. (2024). Changing the dialogue: Descriptive candidacies and position taking in campaigns for the us house of representatives. *Journal of Politics*.
Proksch, S.-O., & Slapin, J. B. (2009). How to avoid pitfalls in statistical analysis of political texts: The case of germany. *German Politics*, *18*(3), 323–344.
Proksch, S.-O., & Slapin, J. B. (2010). Position taking in european parliament speeches. *British Journal of Political Science*, *40*(3), 587–611.
Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. *American journal of political science*, *54*(1), 209–228.
Rodriguez, P. L., & Spirling, A. (2022). Word embeddings: What works, what doesn't, and how to tell the difference for applied research. *The Journal of Politics*, *84*(1), 101–115.
Schulhoff, S., Ilie, M., Balepur, N., Kahadze, K., Liu, A., Si, C., ... others (2024). The prompt report: A systematic survey of prompt engineering techniques. *arXiv preprint arXiv:2406.06608*.
Slapin, J. B., & Proksch, S.-O. (2008). A scaling model for estimating time-series party positions from texts. *American Journal of Political Science*, *52*(3), 705–722.
Tausanovitch, C., & Warshaw, C. (2017). Estimating candidates' political orientation in a polarized congress. *Political Analysis*, *25*(2), 167–187.
Timoneda, J. C., & Vera, S. V. (2025). Bert, roberta, or deberta? comparing performance across transformers models in political science text. *The Journal of Politics*, *87*(1), 347–364.
Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., ... others (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
Westwood, S. J. (2025). The potential existential threat of large language models to online survey research. *Proceedings of the National Academy of Sciences*, *122*(47), e2518075122.
Widmann, T., & Wich, M. (2023). Creating and comparing dictionary, word embedding, and transformer-based models to measure discrete emotions

in german political text. *Political Analysis*, *31*(4), 626–641.

Wong, J. S. H. (2026). Forecasting the use of force: A word embedding analysis of china's rhetoric and military escalations. *Political Science Research and Methods*. (FirstView, published online 23 January 2026)

Wu, X., Wang, M., Liu, Y., Shi, X., Yan, H., Xiangju, L., ... Zhang, W. (2025). Lifbench: Evaluating the instruction following performance and stability of large language models in long-context scenarios. In *Proceedings of the 63rd annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 16445–16468).

Yang, E., Wang, Z., Zhou, C., & Xu, Y. (2025). Data annotation with large language models: Lessons from a large empirical evaluation. *URL: https://www.eddieyang.net/research/llm*$_a$*nnotation.pdf*.

Ying, L., Montgomery, J. M., & Stewart, B. M. (2022). Topics, concepts, and measurement: A crowdsourced procedure for validating topics as measures. *Political Analysis*, *30*(4), 570–589. doi: 10.1017/pan.2021.33

You, K. (2025). Semantics at an angle: When cosine similarity works until it doesn't. *arXiv preprint arXiv:2504.16318*.

Appendix A

Pipeline Implementation

A1 Text inclusion

To isolate issue-relevant text, campaign platforms are first segmented into individual paragraphs. Very short paragraphs of fewer than fifteen words are aggregated into the nearest multi-sentence paragraph, producing a corpus of 179,653 paragraphs. A random sample of 8,262 paragraphs, constituting about 4.6% of the full corpus, was labeled for the presence or absence of content related to our four targeted latent constructs: abortion (pro-life vs. pro-choice), guns (access vs. restriction), healthcare (public option vs. privatization), and immigration (inclusive vs. exclusive). Three trained human readers coded paragraphs for construct-relevant content; coding was not mutually exclusive, as a single paragraph could address multiple constructs. To assess inter-coder reliability, all three readers labeled a common set of 305 paragraphs, representing 4% of the sample, achieving 87% agreement

These human-labeled paragraphs are used to train supervised machine learning models that predict whether a campaign-platform paragraph contains content related to each issue area. Prior to model fitting, paragraphs are pre-processed by converting all text to lowercase and removing stop words and punctuation. Following the recommendations of Park and Montgomery (2025), the 8,262 labeled paragraphs are divided into an 80-20 training-validation split, yielding 6,609 paragraphs for model training and 1,653 for validation.

For each issue, four base classifiers are estimated: a support vector machine, ridge regression, a random forest, and XGBoost. Hyperparameters for all models are selected using a five-fold cross-validated grid search, in which the training data are repeatedly partitioned into held-in and held-out folds, candidate parameter settings are evaluated on the held-out folds, and the specification that performs best on average is retained. Predictions from these base classifiers are then used as inputs to a logistic regression stacking model—an ensemble method that combines base classifier predictions into a single final prediction at a second stage. This approach enables the ensemble to leverage complementary strengths across models, often improving overall performance.

Classifier performance is evaluated by generating predictions for the held-out validation set and comparing them to the human-coded labels. Table 5 reports out-of-sample validation metrics for the resulting stacking classifier. Across all four issue areas, the ensemble achieves strong performance, with F1 scores ranging from 0.824 to 0.929 and consistently high precision and recall. These issue-specific predictions are then applied to all remaining paragraphs in the corpus, classifying each as germane or non-germane to each issue area; predicted-positive paragraphs are aggregated upward to construct alternative text-inclusion units for downstream scaling.

Table 5 Out-of-Sample Validation Metrics for Stacking Classifier

	Accuracy	Precision	Recall	F1
Abortion	0.992	0.852	0.912	0.881
Guns	0.992	0.904	0.955	0.929
Healthcare	0.971	0.752	0.911	0.824
Immigration	0.983	0.893	0.862	0.877

A2 Representation

A2.1 Document-Feature Matrix

For DFM-based pipelines, a common preprocessing routine was applied uniformly across issue areas and text-inclusion levels prior to converting raw platform texts into numerical representations. All text was lowercased, and substantively meaningful multi-word expressions and hyphenated policy terms were normalized to ensure consistent treatment during tokenization. This step preserved phrases whose meaning would otherwise be split across multiple tokens. For example, expressions such as Medicare for All, single-payer, pro-choice, pro-life, and 2nd Amendment were converted into standardized forms (e.g., `medicareforall`, `singlepayer`, `prochoice`, `prolife`, `secondamendment`). Comparable standardizations were applied to other issue-relevant phrases for which punctuation, hyphenation, or spelling variation could fragment substantively equivalent language across distinct features. After this standardization step, non-alphabetic characters were removed, single-character tokens were dropped, and excess whitespace was trimmed.

A separate document-feature matrix was constructed for each issue area at each level of text inclusion. Documents were tokenized into unigrams, standard English stop words were removed, and the remaining tokens were stemmed. Document-feature matrices were constructed with rows corresponding to documents and columns to features, with entries given by term frequencies. For each issue-specific corpus, the vocabulary was further trimmed by removing terms that appeared in fewer than 5% or more than 95% of documents. This eliminates both extremely rare terms, which increase noise and sparsity, and extremely common terms, which provide little leverage for distinguishing positions across documents. Any rows (documents) rendered empty by these transformations were dropped.

A2.2 Embeddings

Static Embeddings - Doc2Vec

For Doc2Vec-based pipelines, a common preprocessing procedure was applied across issue areas and text-inclusion conditions before mapping raw

platform texts into vector representations. All text was first lowercased, and substantively meaningful multi-word expressions and hyphenated policy terms were standardized so that conceptually equivalent language mapped to a common token representation across documents (e.g., Medicare-for-All → `medicareforall`). This step is important as variation in the surface form of politically salient expressions can fragment their distributional signal, thereby degrading the quality of the learned representations. After this standardization, we removed non-alphabetic characters, dropped single-character tokens, and trimmed excess whitespace.

A Doc2Vec model is then estimated that jointly learns embeddings for words and for document identifiers attached to the cleaned platform texts. In our application, these identifiers are assigned at the candidate–issue–year level, though the exact indexing scheme depends on the level of textual aggregation. When the model is trained on paragraphs or platform points, multiple relevant texts may share the same candidate–issue–year identifier, thus each contributes to the estimation of a common embedding for that unit. Paragraphs and platform points classified as non-relevant to the target construct are still supplied to the model, but without a candidate–issue–year identifier, thereby contributing only to word embedding estimation. When the model is trained on full campaign platforms, a single document often contributes to multiple candidate–issue–year observations. In these cases, the platform text is duplicated in the training input and paired with each relevant document identifier, allowing its semantic content to contribute to every corresponding unit-level embedding. The same procedure is applied whenever a platform point or paragraph is relevant to more than a single candidate–issue–year combination.

The Doc2Vec model was initialized with pre-trained word embeddings from the Google News corpus, with random starting values assigned to the candidate–issue–year vectors. A paragraph vector distributed bag-of-words (PV-DBOW) model was fit to the campaign-policy corpus using 300-dimensional embeddings, a context window of six tokens, and a minimum term-frequency threshold of 10 to exclude very rare words, consistent with the recommendations of Rodriguez and Spirling (2022). Word embeddings were permitted to update during estimation, consistent with the recommendations of Lau and Baldwin (2016) and Case (2027). The model returned a single embedding for each candidate–issue–year unit, capturing the semantic content of rhetoric associated with a given candidate on a given issue in a given election year—shaped both by the distributional structure encoded in the pre-trained word vectors and by further tuning to the issue-specific campaign text.

Contextualized Embeddings - RoBERTa, Nomic, and OpenAI

For the contextualized embedding pipelines produced with RoBERTa, Nomic Embed, and OpenAI Embeddings, the text standardization and cleaning procedures used in the DFM and Doc2Vec approaches were not applied. Raw

policy text was retained instead, as punctuation, casing, and local word order all contribute to contextual meaning that these models are designed to capture.

Contextualized embedding models map an input token sequence to context-sensitive numerical representations. The maximum length of text the model can encode in a single pass is fixed by the model itself. When a given policy document (paragraph, platform point, or platform) exceeded the maximum input sequence length, we partitioned it into segments no longer than the model's limit. We first split documents into sentence-like units using sentence-ending punctuation and line breaks as boundaries.[32] Sentence units were then recombined into segments subject to the model's token limit, ensuring each segment remained within the allowable input length. This preserved local semantic and syntactic context by avoiding arbitrary mid-sentence breaks during segmentation. In practice, segmentation was uncommon for paragraph-level inputs, but it occurred much more frequently at the full-platform level. No stand-alone sentences were longer than the maximum token input.

We provide more detailed model specifications for producing and extracting embeddings with RoBERTa, Nomic, and OpenAI models below.

RoBERTa

We estimated embeddings for RoBERTa pipelines using the `all-roberta-large-v1` model through the `SentenceTransformer` library in Python. We use the model's maximum input sequence length of 256 tokens, consistent with the default configuration of the SentenceTransformers implementation of `all-roberta-large-v1`. Inputs exceeding this length were segmented using the procedure described above. The RoBERTa model maps each input sequence to a 1,024-dimensional vector representation. Our RoBERTa implementation returns a pooled sequence-level embedding for each input text. When multiple encoded segments mapped to the same candidate–issue–year unit, we combined them using a token-length-weighted average, so that longer segments contributed proportionally more to the unit's final representation. We then re-normalized each aggregated unit-level embedding to unit length. This procedure yielded a single embedding for each candidate-issue-year unit.

Nomic Embed

We estimated embeddings for Nomic pipelines using the `nomic-ai/nomic-embed-text-v1` model through the `SentenceTransformer` library in Python. Following Nomic's recommended document-formatting procedure, we encoded each text segment with the prefix `search_document:`, which conditions the model to encode the input as a document rather than as a short query, yielding vector representations better suited to document-level semantic similarity. When the text input for a given candidate–issue–year unit exceeded the model's maxi-

[32]Common abbreviations and patterned forms (e.g., U.S., Dr., Sen.) were ignored so that punctuation internal to abbreviations would not produce inaccurate sentence breaks.

mum input sequence length of 8,192 tokens, we applied the text segmentation steps described above. Nomic Embed returns a single dense embedding vector for each encoded segment. Each segment embedding is 768-dimensional and normalized to unit length, allowing segment-level similarity to be computed directly using cosine similarity or the dot product. When multiple encoded segments mapped to the same candidate–issue–year unit, we combined them using a token-length-weighted average, so that longer segments contributed proportionally more to the unit's final representation. This procedure yielded a single embedding for each candidate-issue-year unit.

OpenAI

We estimated embeddings for OpenAI pipelines using the `text-embedding-3-large` model through the OpenAI API. When the text input for a given candidate–issue–year unit exceeded the model's maximum input sequence length of 8,192 tokens, we applied the text segmentation steps described above. The OpenAI model version we employ returns a single 3,072-dimensional vector representation for each encoded segment. When multiple encoded segments mapped to the same candidate–issue–year unit, we combined them using a token-length-weighted average, so that longer segments contributed proportionally more to the unit's final representation. We then re-normalized each aggregated unit-level embedding to unit length. This procedure yielded a single embedding for each candidate–issue–year unit.

A2.3 Large Language Model

For the large language model pipelines, we follow the same basic preprocessing logic used in the OpenAI embedding approach. Rather than applying the more aggressive standardization and cleaning procedures used in the DFM and Doc2Vec pipelines, we retain the raw policy text in its original form. This choice reflects the fact that generative language models rely on punctuation, capitalization, syntax, and local word order when interpreting meaning. Preserving the original text therefore allows the model to evaluate candidate statements using the fuller set of contextual cues contained in the source material.

Our LLM-based scoring uses the `meta-llama/llama-3.3-70b-instruct` model accessed through OpenRouter's chat-completions API. Across pipelines, the model is applied to the combined issue-specific text units linked to the relevant candidate. Depending on the pipeline, these units may consist of individual paragraphs, complete issue statements, or broader platform text. In each case, the model receives the relevant text unit as input and returns a short structured output that can be converted into an issue-specific score. To keep responses stable across observations, we set `temperature = 0`. A key advantage of this approach is that the pipeline can accommodate the full text associated with each unit of analysis. As a result, the LLM pipelines do not require the kinds of truncation or segmentation decisions that arise in some

other text representation strategies.

Across LLM pipelines and issue areas, we develop a general prompt structure relevant at the representation step. The prompt structure serves two main purposes. First, it defines the relevant policy domain so that the model evaluates only the issue content of interest and ignores unrelated material. In practice, this means that the prompts identify the issue area under consideration, clarify what kinds of content should count as relevant to that issue, provide general guidance for handling mixed, ambiguous, or weakly expressed positions. Second, provides a numerical scale (0 to 100) for scoring texts. This second piece follows the same prompting logic from Le Mens and Gallego (2025). An example of the general prompting structure reads as follows:

> You are scoring a U.S. congressional candidate statement on abortion policy. Score the statement's abortion-policy position on a 0–100 very left to very right scale.
>
> DEFINITION (Abortion policy): Statements about access to abortion, abortion bans or restrictions, Roe v. Wade / Dobbs, federal or state abortion legislation, exceptions (rape, incest, life of the mother), gestational limits, abortion as a right or as murder, and Planned Parenthood ONLY when explicitly tied to abortion. Do NOT count contraception, reproductive education, women's healthcare, or Planned Parenthood unless abortion is explicitly discussed.
>
> SCORING PRINCIPLES:
>
> - Score ONLY the abortion-related content; ignore unrelated content.
>
> - Score what the text says explicitly. Do not infer positions.
>
> - Use the FULL range, but 0 and 100 should be RARE.
>
> - If the text is mixed or balanced, pull toward the center (around 50).
>
> - If the text expresses a direction weakly or rhetorically, use scores near the midpoints.
>
> - If the text discusses abortion but does not take a clear pro-choice or pro-life stance, score as ambiguous (near 50).
>
> OUTPUT FORMAT: Return ONLY valid JSON: "Score": <number 0-100> or "Score":"NA" if the text is irrelevant to abortion policy. Do not include any explanations. If the text discusses abortion but takes no explicit stance, score the statement as a 50.

The supervised and unsupervised variants differ in the extent to which they provide additional guidance for mapping text onto the underlying scale. We provide more details on this in the next section, as well as full prompts.

A3 Scaling Procedure

A3.1 Document-Feature Matrix

For unsupervised scaling, we generate positional measures using a standard Poisson Wordfish model (Slapin & Proksch, 2008), estimating separate models for each issue-specific DFM described above. Wordfish measures, π_i^{WF}, denote a scalar position for each candidate–issue–year representation on a single latent dimension of term discrimination in the corpus. Because Wordfish does not identify the polarity of the latent dimension, its direction is arbitrary. We therefore reoriented our measures based on the estimated mean positions of Democratic and Republican candidates. When the Democratic mean exceeded the Republican mean, we multiplied all scores for that issue by -1. This adjustment preserves relative distances among candidates while ensuring a consistent interpretation across pipeline configurations. Following reorientation, we rescaled all measures to have a mean of zero and unit variance to additionally facilitate comparison across pipeline configurations.

For the weakly supervised scaling, we generate positional measures using Wordscores (Laver et al., 2003), estimating separate models for each issue-specific DFM described above. To anchor each issue-specific left-right dimension, we generate synthetic reference documents using OpenAI's GPT. Specifically, we prompt GPT to generate campaign-style policy statements spanning five ordered categories (very left, left, moderate, right, and very right) using issue-specific codebooks to define the substantive content associated with each position. These codebooks align with the instructions provided to human readers for validation, as described in Appendix B1. For each issue area, we generate 20 statements at each extreme (very left/right) and 10 in each intermediate category, placing greater density at the endpoints to ensure a clearer definition of the scale's extremes. This quantity and distribution of reference documents follows the guidance of Laver et al. (2003) for Wordscore implementation. We prompted GPT to constrain generated texts to the average length of campaign platform points, about 110-130 words. We then manually reviewed the synthetic texts to confirm that they reflected the intended positions and did not contain misaligned content. We then converted the synthetic reference corpus to a DFM using the same preprocessing procedures applied to the policy platform texts.

We fit Wordscores models to reference document DFMs, assigning each row a fixed reference value on a five-point scale from -1 (very left) to 1 (very right) based on the document's designated position. We applied additive smoothing during estimation to improve stability, given sparse term counts. After fitting the Wordscores model for each issue area, we applied the fitted model to the corresponding DFM to predict issue-specific positional measures, $\hat{\pi}_i^{\text{WS}}$, for candidate–issue–year units. The resulting measures were signed such that larger values corresponded to more right-leaning positions and smaller values to more left-leaning positions, aligning with the anchored directionality of the issue scale. Accordingly, no polarity correction was required. Finally, we rescaled scores

Table 6 Positional Term Dictionaries for Semantic Projection

Issue Area	Left-Most Position	Right-Most Position
Abortion	**Pro-Choice**: reproductive, justice, freedom, health, care, healthcare, expand, access, safe, hyde, roe, wade, choice, prochoice, government, codifying, legal, restrictive, decisions, autonomy, control, privacy, doctor, respect, personal, equality, women, sexual, services	**Pro-Life:** life, birth, death, unborn, womb, precious, sanctity, moment, conception, heartbeat, begins, prolife, partial, fetal, born, alive, demand, saving, defund, taxpayer, infanticide, murder, barbaric, innocent, dignity, values, conscientious, immoral, vulnerable, defenseless, ban, outlaw, god, gift, sacred
Guns	**Increase Restrictions:** mandatory, background, ban, national, hate, registry, database, assault, automatic, ar, ak, military, battlefield, war, waiting, age, years, ghost, loopholes, transfer, traumatic, violence, mass, killing, crisis, epidemic, dealers, manufacturers, trafficking, comprehensive, stricter	**Increase Access:** second, amendment, owner, abiding, right, infringed, inherent, unconstitutional, founding, enumerated, confiscation, unrestricted, repeal, oppose, abolish, any, overturn, enshrined, reciprocity, concealed, carry, hearing, security, freedoms, fundamental, defending, families, protection
Health Insurance	**Publicly Funded:** universal, all, deny, coverage, privilege, guarantee, uninsured, gap, public, medicareforall, singlepayer, free, strengthen, expand, comprehensive, dental, hearing, vision, access, human, right, equality, fundamental, poor, justice, inclusive, profit, preexisting, bankruptcy	**Privatize:** repeal, replace, problems, flawed, free, market, open, competition, freedom, national, state, choice, consumers, individual, patient, provider, insurance, mandate, Washington, burdensome, tape, involvement, bureaucratic, deregulate, tax, hsa, savings, obamacare, socialized
Immigration	**Inclusive:** undocumented, pathway, roadmap, temporary, tps, dreamers, daca, asylum, humane, compassion, dignity, diverse, culture, xenophobic, racist, families, reunification, detention, cages, humanitarian, streamline, backlogs, courts, accessible, opportunity, expand, abolish, ice, education, healthcare	**Exclusive:** enforce, rule, law, secure, build, wall, barrier, surveillance, technology, verify, screening, safety, protect, sanctuary, chain, migration, anchor, birthright, english, skilled, merit, labor, overstay, deport, amnesty, illegal, criminal, terrorists, flooding, porous, close, funding, hire, officers, patrol

within each issue to have a mean of zero and unit variance for comparability across pipeline configurations.

A3.2 Embeddings

For unsupervised scaling, we generate positional measures using principal component analysis (PCA). For all embedding representations, we apply PCA to the issue-specific embedding matrix and retain the first principal component. The resulting PCA measure, $\hat{\pi}_i^{\text{PCA}}$, places each candidate–issue–year unit on the dominant dimension of semantic variation within that issue area. Because the sign of the first principal component is not identified, its direction is arbitrary. We therefore reorient the recovered scores using the estimated mean positions of Democratic and Republican candidates. When the Democratic mean exceeds the Republican mean, we multiply all scores for that issue by -1. This adjustment preserves relative distances among candidates while ensuring a consistent interpretation across pipeline configurations. Following reorientation, we rescale all measures to have a mean of zero and a unit variance to additionally facilitate comparison across pipeline configurations.

For the weakly supervised scaling, we generate positional measures via semantic projection. To anchor targeted issue-specific dimensions using semantic projection, we specify term lists intended to capture the left-most (L) and right-most positions for each issue-specific conflict dimension, denoted by the low-pole (L) and high-pole (H), respectively. These term lists were constructed exogenously to the CampaignView using issue-aligned PAC position-taking

statements to identify substantively appropriate endpoint language. A full list of the terms defining these issue-specific poles is shown in Table 6. We defined the two poles as the mean embedding (μ) for each term list, $L = \{\ell_1, \ldots, \ell_{t_L}\}$ and $H = \{h_1, \ldots, h_{t_H}\}$, and the semantic axis as the normalized difference between the high-endpoint mean vector, μ_H, and the low-endpoint mean vector, μ_L. We then scaled positions, $\hat{\pi}_i^{\text{SP,magnitude}}$, by projecting each candidate–issue–year embedding onto the semantic axis, computed as the embedding norm multiplied by its cosine similarity with that axis.

The exact implementation of semantic projection varies. For Doc2Vec pipelines, word embeddings are jointly learned with the candidate–issue–year embeddings; we extract the embeddings associated with each word in the term list to construct the mean embedding for the pole indicative of left-most position (L) and right-most position (H). For the remaining embedding pipelines (RoBERTa, Nomic Embed, and OpenAI Embeddings), embeddings are generated only for sequence-level inputs, so these models do not produce standalone word vectors. Accordingly, we represent endpoint terms in context rather than in isolation by embedding each keyword within a short issue-specific prompt (e.g., "The following word concerns [issue] policy: [keyword]."). This procedure yields text-level representations of the endpoint terms in a policy-relevant semantic frame, which we then use to construct the corresponding pole embeddings.

A3.3 Large Language Model

For unsupervised scaling, we generate positional measures directly from the large language model using issue-specific prompts that do not provide a codebook or explicit substantive calibration of the left–right scale. Instead, the prompt identifies the policy domain under consideration, instructs the model to evaluate only the relevant issue content, and asks it to return a position on a 0–100 scale from very left to very right. In this configuration, the model must infer the substantive meaning of the issue dimension from its internal knowledge and from the text itself rather than from externally supplied position anchors. The resulting measure is therefore unsupervised in the sense that the researcher does not provide exemplars or a codebook specifying what should count as left, right, or moderate within the issue area. At the same time, because the prompt explicitly defines the direction of the scale from very left to very right, polarity is fixed ex ante and no post hoc reorientation is required. As with the other pipeline configurations, we then rescale the recovered measures within each issue to have a mean of zero and unit variance for comparability across pipeline configurations.

For the weakly supervised scaling, we also generate positional measures directly from the large language model, but here the prompt includes issue-specific codebook guidance to anchor the underlying policy dimension. Rather than asking the model to infer the meaning of left and right entirely on its own, we provide additional substantive direction about the kinds of positions,

proposals, and rhetorical cues that correspond to different locations on the scale. In practice, this means that the prompt not only defines the relevant policy domain and the required output format, but also supplies codebook-aligned descriptions of what constitutes more left-leaning, moderate, and more right-leaning issue content. The resulting measure is weakly supervised because of the supplied structured substantive input to guide how the model maps text onto the issue dimension, while the model still performs the final judgment for each text unit. As in the unsupervised LLM pipelines, the requested output is already oriented from very left to very right, so no polarity correction is necessary. We again rescale scores within an issue to have a mean of zero and unit variance to facilitate comparison across pipeline configurations.

A3.4 Full LLM Prompt Language

Below, we provide the full text for all of our LLM prompts across the four policy areas. For unsupervised prompting, we remove the scoring calibration section from each prompt and use the remaining text as the prompt to the LLM.

Abortion:
You are scoring a U.S. congressional candidate statement on abortion policy.

Task: Score the statement's abortion-policy position on a 0–100 very left to very right scale.

DEFINITION (Abortion policy):
Statements about access to abortion, abortion bans or restrictions, Roe v. Wade / Dobbs, federal or state abortion legislation, exceptions (rape, incest, life of the mother), gestational limits, abortion as a right or as murder, and Planned Parenthood ONLY when explicitly tied to abortion. Do NOT count contraception, reproductive education, women's healthcare, or Planned Parenthood unless abortion is explicitly discussed.

SCORING PRINCIPLES:
- Score ONLY the abortion-related content; ignore unrelated content.
- Score what the text says explicitly. Do not infer positions.
- Use the FULL range, but 0 and 100 should be RARE.
- If the text is mixed or balanced, pull toward the center (around 50).
- If the text expresses a direction weakly or rhetorically, use scores near the midpoints.
- If the text discusses abortion but does not take a clear pro-choice or pro-life stance, score as ambiguous (near 50).

Scoring Calibration:
0–15 (very left): Abortion access without restrictions; abortion framed as a

fundamental right; calls to codify Roe v. Wade; repeal Hyde Amendment or global gag rule; "abortion is healthcare," "reproductive justice," "reproductive rights."
20–35 (left): General pro-choice rhetoric; protecting women's right to choose; abortion should be available; may express personal opposition but supports access; no explicit legislative detail.
40–60 (centrist/ambiguous): Abortion available with restrictions (e.g., trimester or 20-week limits); balanced mix of pro-choice and pro-life language; emphasis on reducing abortions without banning them; discussion of adoption or education without a clear policy stance.
65–80 (right): Pro-life rhetoric; abortion unlawful in most cases but with common exceptions (rape, incest, child pregnancy, life of the mother); "right to life," "life is precious."
85–100 (very right): No abortions under any circumstances except possibly life of the mother; abortion framed as murder or homicide; "life begins at conception"; calls for federal bans or outlawing abortion nationwide.

OUTPUT FORMAT:
Return ONLY valid JSON: {"Score": <number 0-100>} or {"Score":"NA"} if the text is irrelevant to abortion policy. Do not include any explanations. If the text discusses abortion but takes no explicit stance, score the statement as a 50.

Guns:
You are scoring a U.S. congressional candidate statement on gun policy.

Task: Score the statement's gun-policy position on a 0–100 very left to very right scale.

DEFINITION (Gun policy — restriction vs. access):
Statements about gun access, gun control, and firearms regulation, including: bans or restrictions on certain weapons (assault weapons, military-style weapons), background checks, waiting periods, age limits, permitting/licensing, national registries, gun manufacturer liability, red flag laws, gun-free zones, concealed carry laws, reciprocity across states, guns in schools, and explicit support for or opposition to the Second Amendment as it relates to regulation.

SCORING PRINCIPLES:
- Score ONLY the gun-related content; ignore unrelated content.
- Score what the text says explicitly. Do not infer positions.
- Use the FULL range, but 0 and 100 should be RARE.
- If the text is mixed or balanced, pull toward the center (around 50).
- If the text expresses a direction weakly or rhetorically, use scores near the midpoints.
- If the text discusses guns but does not take a clear stance on restriction vs.

access, score as ambiguous (near 50).

Scoring Calibration:
0–15 (very left): Strong restrictions on guns themselves; bans on assault weapons or military-style firearms; repeal legal protections for gun manufacturers/dealers; bans on most or all weapons.
20–35 (left): Some restrictions on purchase and access; waiting periods; raising age limits; limits on where/when guns can be purchased; assault-weapon permitting; national gun registry; generally "stricter gun laws."
40–60 (centrist/ambiguous): "Common-sense" gun reforms; universal background checks; red flag laws; closing loopholes; banning bump stocks or high-capacity magazines; buy-back programs with the option to keep guns; pro–Second Amendment rhetoric paired with explicit restrictions.
65–80 (right): Emphasizes Second Amendment rights; opposes additional gun restrictions; rhetoric about preventing gun confiscation; supports gun ownership with minimal discussion of regulation.
85–100 (very right): No limits on gun access; repeal restrictions on concealed carry; reciprocity across states; guns allowed in schools or campuses; opposition to background checks or waiting periods; explicit rejection of any gun control laws.

OUTPUT FORMAT:
Return ONLY valid JSON: {"Score": <number 0-100>} or {"Score":"NA"} if the text is irrelevant to gun policy (as defined above). Do not include any explanations. If the text discusses guns but takes no explicit stance, score the statement as a 50.

Healthcare:
You are scoring a U.S. congressional candidate statement on healthcare policy.

Task: Score the statement's healthcare-policy position on a 0–100 very left to very right scale.

DEFINITION (Healthcare policy — health insurance marketplace):
Statements about the role of government vs. private markets in health insurance and coverage, including: single-payer/universal healthcare/Medicare-for-all, public option, expanding Medicaid/Medicare, protecting/maintaining the ACA/Obamacare, repealing the ACA/Obamacare, free-market reforms (competition, purchasing across state lines), and the degree of government involvement in the insurance marketplace.

SCORING PRINCIPLES:
- Score ONLY the healthcare-related content; ignore unrelated content.
- Score what the text says explicitly. Do not infer positions.

- Use the FULL range, but 0 and 100 should be RARE.
- If the text is mixed or balanced, pull toward the center (around 50).
- If the text expresses a direction weakly or rhetorically, use scores near the midpoints.
- If the text discusses healthcare but does not take a clear stance on government vs. market role in insurance, score as ambiguous (near 50).

Scoring Calibration:
0–15 (very left): Full government involvement in health insurance; single-payer/universal system; "Medicare-for-all"; abolish or replace private insurance; healthcare framed as a fundamental human right.
20–35 (left): Expanded government involvement but not full single-payer; expand public options; expand Medicaid; expand Medicare; may explicitly keep private insurance for those who want it.
40–60 (centrist/ambiguous): Maintain/protect the status quo; protect ACA; protect Medicare; protect Medicaid; or mixed language that does not clearly increase or decrease government involvement; broad affordability/access rhetoric with no clear mechanism.
65–80 (right): Little government involvement; emphasizes choice and market competition; purchasing across state lines; competition/free-market language; may still support limited protections (e.g., preexisting condition coverage, cost caps) while emphasizing markets.
85–100 (very right): Remove government from the health insurance marketplace; fully free-market system; repeal Obamacare/ACA and replace with competitive market-based system; strong opposition to government involvement.

OUTPUT FORMAT:
Return ONLY valid JSON: {"Score": <number 0-100>} or {"Score":"NA"} if the text is irrelevant to healthcare policy (as defined above). Do not include any explanations. If the text discusses healthcare but takes no explicit stance, score the statement as a 50.

Immigration:
You are scoring a U.S. congressional candidate statement on immigration policy.

Task: Score the statement's immigration-policy position on a 0–100 very left to very right scale.

DEFINITION (Immigration policy — inclusive vs. exclusive):
Statements about treatment of immigrants (documented or undocumented), pathways to citizenship (including DACA, TPS, earned pathways), deportation, border control and enforcement (ICE, border security, wall), asylum, family reunification, birthright citizenship, limits on who can immigrate, and the overall inclusiveness vs. restrictiveness of immigration policy.

SCORING PRINCIPLES:
- Score ONLY the immigration-related content; ignore unrelated content.
- Score what the text says explicitly. Do not infer positions.
- Use the FULL range, but 0 and 100 should be RARE.
- If the text is mixed or balanced, pull toward the center (around 50).
- If the text expresses a direction weakly or rhetorically, use scores near the midpoints.
- If the text discusses immigration but does not take a clear stance on inclusiveness vs. restrictiveness, score as ambiguous (near 50).

Scoring Calibration:
0–15 (very left): Highly inclusive immigration policy; pathway to citizenship for all undocumented immigrants; end deportations; abolish or demilitarize ICE; provide healthcare, education, and humane treatment regardless of status; maximize access to immigration.
20–35 (left): Inclusive but conditional pathways to citizenship (e.g., DACA-only, TPS, earned pathways); humane border treatment; opposition to family separation; mix of law enforcement with strong emphasis on access and legalization.
40–60 (centrist/ambiguous): Even mix of border enforcement and some pathway to citizenship; balanced law-and-compassion rhetoric; or mixed language without clear emphasis on inclusion or exclusion.
65–80 (right): Emphasis on lawful immigration; prioritize border security; require undocumented immigrants to follow regular processes; limited or selective pathways; reduce overall access to immigration while recognizing some labor needs.
85–100 (very right): Highly restrictive policy; all undocumented immigrants treated as criminals; no amnesty; mass deportations; end birthright citizenship; severe limits on who can immigrate (e.g., bans, country-specific restrictions); close borders or end most immigration.

OUTPUT FORMAT:
Return ONLY valid JSON: {"Score": <number 0-100>} or {"Score":"NA"} if the text is irrelevant to immigration policy (as defined above). Do not include any explanations. If the text discusses immigration but takes no explicit stance, score the statement as a 50.

Appendix B

Measurement Validation

B1 Construct Validation

In this validation exercise, two undergraduate student readers and one principal investigator (PI) independently scored the same random sample of 900 documents. Each document was a paragraph-level text previously classified as relevant to one of the issue areas under study (see Appendix B.1). Readers assigned a single score to the full set of paragraphs associated with each candidate–issue–year observation, using a five-point integer scale ranging from very left (−2) to very right (+2). Texts flagged as ambiguous were assigned a value of zero (i.e., centrist); raters identified ambiguity in roughly 10% of sampled documents. Table 7 reports inter-coder agreement in scoring policy platform paragraphs; for most texts, at least two readers assigned the same score.

Table 7 Reader Agreement Rate, By Issue Area

Issue Area	3/3 Agreement	2/3 Agreement	No Agreement
Abortion	0.675	0.322	0.003
Guns	0.554	0.446	0
Healthcare	0.551	0.425	0.024
Immigration	0.486	0.510	0.003

B2 Predictive Validation

We estimate a series of logistic regressions in which receipt of a contribution from an issue-aligned PAC is modeled as a function of the extremity of a candidate's scaled position on the corresponding issue dimension. The outcome includes all PAC support for a candidate, including both direct contributions and independent expenditures. We analyze contribution receipt rather than contribution amount because funding levels are more likely to be endogenous to factors beyond the candidate's issue position, such as district competitiveness and broader electoral conditions. We also restrict the analysis to candidates who won their primary and appeared in the general election, as PAC giving in primaries is relatively infrequent. Issue-aligned PACs for the abortion and gun dimensions are identified using OpenSecrets. In the abortion and gun analyses, the included PACs are ideological or single-issue organizations that advocate clear positions on the relevant policy dimension. Operationally, the analysis is limited to contributions from left-leaning PACs to Democratic candidates and from right-leaning PACs to Republican candidates.

Table 8 Abortion PAC Donation (DFM)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.87*	0.41*	0.72*	0.38*	0.74*	0.04
	(0.15)	(0.12)	(0.13)	(0.12)	(0.13)	(0.08)
Republican	0.42*	0.41*	0.47*	0.40*	0.57*	0.42*
	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)
Constant	−0.08	0.39*	0.09	0.45*	0.06	0.68*
	(0.15)	(0.13)	(0.14)	(0.12)	(0.14)	(0.09)
Observations	1,096	1,096	1,096	1,096	1,096	1,096

Note: *p<0.05

Table 9 Abortion PAC Donation (Doc2Vec)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	1.07*	0.59*	1.02*	0.50*	1.45*	0.19*
	(0.14)	(0.12)	(0.14)	(0.12)	(0.18)	(0.09)
Republican	0.46*	0.41*	0.52*	0.41*	0.35*	0.44*
	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)
Constant	−0.29	0.20	−0.29	0.28*	−0.57*	0.58*
	(0.15)	(0.13)	(0.16)	(0.13)	(0.18)	(0.10)
Observations	1,096	1,096	1,096	1,096	1,096	1,096

Note: *p<0.05

Table 10 Abortion PAC Donation (RoBERTa)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.47*	0.19	0.55*	0.14	0.23*	0.02
	(0.09)	(0.12)	(0.09)	(0.12)	(0.07)	(0.09)
Republican	0.53*	0.42*	0.58*	0.42*	0.54*	0.42*
	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)
Constant	0.36*	0.55*	0.30*	0.59*	0.61*	0.69*
	(0.11)	(0.12)	(0.11)	(0.12)	(0.09)	(0.10)
Observations	1,096	1,096	1,096	1,096	1,096	1,096

Note: *p<0.05

Table 11 Abortion PAC Donation (Nomic)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.98*	0.25*	0.78*	0.11	0.41*	−0.28*
	(0.14)	(0.11)	(0.11)	(0.11)	(0.09)	(0.07)
Republican	0.47*	0.41*	0.56*	0.41*	0.56*	0.43*
	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)
Constant	−0.19	0.50*	0.01	0.61*	0.41*	0.76*
	(0.15)	(0.12)	(0.13)	(0.12)	(0.11)	(0.09)
Observations	1,096	1,096	1,096	1,096	1,096	1,096

Note: *p<0.05

Table 12 Abortion PAC Donation (OpenAI)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	1.22*	0.61*	1.07*	0.52*	0.44*	0.31*
	(0.17)	(0.15)	(0.14)	(0.15)	(0.10)	(0.10)
Republican	0.44*	0.43*	0.53*	0.44*	0.51*	0.46*
	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)	(0.14)
Constant	−0.44*	0.15	−0.29	0.24	0.35*	0.49*
	(0.18)	(0.16)	(0.16)	(0.16)	(0.11)	(0.11)
Observations	1,096	1,096	1,096	1,096	1,096	1,096

Note: *p<0.05

Table 13 Abortion PAC Donation (LLaMA)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.31*	1.76*	0.38*	1.51*	0.04	0.96*
	(0.11)	(0.29)	(0.11)	(0.29)	(0.13)	(0.35)
Republican	0.39*	0.24	0.36*	0.25	0.45*	0.33*
	(0.14)	(0.15)	(0.15)	(0.15)	(0.15)	(0.16)
Constant	0.50*	−0.84*	0.46*	−0.59*	0.71*	−0.11
	(0.12)	(0.27)	(0.12)	(0.28)	(0.15)	(0.33)
Observations	1,002	1,003	969	1,021	953	871

Note: *p<0.05

Table 14 Guns PAC Donation (DFM)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.81*	0.46*	0.81*	0.44*	0.36*	0.05
	(0.12)	(0.10)	(0.12)	(0.10)	(0.11)	(0.07)
Republican	1.46*	1.38*	1.45*	1.37*	1.45*	1.30*
	(0.13)	(0.13)	(0.13)	(0.13)	(0.13)	(0.12)
Constant	−1.15*	−0.79*	−1.12*	−0.75*	−0.74*	−0.43*
	(0.14)	(0.12)	(0.13)	(0.12)	(0.13)	(0.09)
Observations	1,213	1,213	1,213	1,213	1,213	1,213

Note: *p<0.05

Table 15 Guns PAC Donation (Doc2Vec)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.88*	0.76*	0.87*	0.67*	1.71*	0.25*
	(0.11)	(0.10)	(0.11)	(0.10)	(0.16)	(0.08)
Republican	1.54*	1.41*	1.50*	1.38*	1.67*	1.32*
	(0.13)	(0.13)	(0.13)	(0.13)	(0.14)	(0.12)
Constant	−1.31*	−1.08*	−1.26*	−0.98*	−2.11*	−0.57*
	(0.15)	(0.13)	(0.14)	(0.12)	(0.19)	(0.10)
Observations	1,213	1,213	1,213	1,213	1,213	1,213

Note: *p<0.05

Table 16 Guns PAC Donation (RoBERTa)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.25*	0.35*	0.22*	0.37*	−0.13*	−0.01
	(0.08)	(0.09)	(0.07)	(0.09)	(0.06)	(0.08)
Republican	1.37*	1.35*	1.36*	1.35*	1.22*	1.28*
	(0.13)	(0.12)	(0.13)	(0.12)	(0.13)	(0.12)
Constant	−0.58*	−0.68*	−0.53*	−0.68*	−0.35*	−0.39*
	(0.10)	(0.11)	(0.09)	(0.11)	(0.09)	(0.09)
Observations	1,213	1,213	1,213	1,213	1,213	1,213

Note: *p<0.05

Table 17 Guns PAC Donation (Nomic)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.79*	0.56*	0.70*	0.47*	0.16*	0.08
	(0.11)	(0.10)	(0.10)	(0.10)	(0.08)	(0.07)
Republican	1.47*	1.36*	1.45*	1.33*	1.36*	1.30*
	(0.13)	(0.12)	(0.13)	(0.12)	(0.13)	(0.12)
Constant	−1.14*	−0.88*	−1.03*	−0.78*	−0.52*	−0.44*
	(0.14)	(0.12)	(0.13)	(0.12)	(0.10)	(0.09)
Observations	1,213	1,213	1,213	1,213	1,213	1,213

Note: *p<0.05

Table 18 Guns PAC Donation (OpenAI)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	0.96*	0.58*	0.84*	0.49*	0.20*	−0.03
	(0.13)	(0.11)	(0.12)	(0.11)	(0.08)	(0.09)
Republican	1.50*	1.37*	1.46*	1.34*	1.36*	1.28*
	(0.13)	(0.13)	(0.13)	(0.12)	(0.13)	(0.12)
Constant	−1.33*	−0.91*	−1.19*	−0.81*	−0.57*	−0.37*
	(0.16)	(0.13)	(0.15)	(0.12)	(0.11)	(0.10)
Observations	1,213	1,213	1,213	1,213	1,213	1,213

Note: *p<0.05

Table 19 Guns PAC Donation (LLaMA)

	Dependent variable: PAC Donation					
Text Inclusion	Paragraph		Platform Point		Platform	
Supervision	U	S	U	S	U	S
Extremity	1.46*	1.61*	1.69*	1.54*	1.63*	1.57*
	(0.22)	(0.20)	(0.23)	(0.21)	(0.26)	(0.25)
Republican	1.62*	1.68*	1.68*	1.57*	1.59*	1.61*
	(0.15)	(0.14)	(0.15)	(0.14)	(0.14)	(0.14)
Constant	−1.96*	−2.08*	−2.21*	−1.97*	−2.12*	−2.07*
	(0.26)	(0.24)	(0.27)	(0.24)	(0.30)	(0.28)
Observations	1,052	1,143	1,088	1,136	1,115	1,066

Note: *p<0.05

B3 Face Validation

We implement face-validation analyses using the `quanteda` package in `R` (Benoit et al., 2018). Keyness statistics are computed with `quanteda`'s `textstat_keyness()` function, which identifies terms that occur disproportionately often in a target set of texts relative to a reference set. Following this implementation, term distinctiveness is evaluated using the χ^2 statistic, so that higher positive values indicate words that more strongly differentiate the target texts from the reference corpus. The main body text describes how we leverage keyness analysis to conduct our face validity tests.