Advance Access publication October 4, 2008

Political Analysis (2008) 16:356–371
doi:10.1093/pan/mpn004

Understanding Wordscores

Will Lowe
Methods and Data Institute, School of Politics and International Relations, University of Nottingham, Nottingham, NG7 2RD, UK
e-mail: will.lowe@nottingham.ac.uk

Wordscores is a widely used procedure for inferring policy positions, or scores, for new documents on the basis of scores for words derived from documents with known scores. It is computationally straightforward, requires no distributional assumptions, but has unresolved practical and theoretical problems. In applications, estimated document scores are on the wrong scale and the theoretical development does not specify a statistical model, so it is unclear what assumptions the method makes about political text and how to tell whether they fit particular text analysis applications. The first part of the paper demonstrates that badly scaled document score estimates reflect deeper problems with the method. The second part shows how to understand Wordscores as an approximation to correspondence analysis which itself approximates a statistical ideal point model for words. Problems with the method are identified with the conditions under which these layers of approximation fail to ensure consistent and unbiased estimation of the parameters of the ideal point model.

1 Introduction

Wordscores (Benoit and Laver 2003; Laver et al. 2003) is a pioneering method of automated content analysis that assigns policy positions or ''scores'' to documents on the basis of word counts and known document scores via the computation of ''wordscores.'' The method is straightforward to implement, requires no functional or distributional assumptions, and works well in many applications (e.g., Benoit and Laver 2003; Klemmensen et al. 2007). It also has some more troubling features: estimated document scores are not directly interpretable without rescaling, and it is often unclear how best to choose a suitable rescaling method. Wordscores is also expressed directly as an algorithm rather than being derived from an underlying model. In the absence of a statistical model, it is unclear what assumptions Wordscores makes about the relationship between document scores and words, so it is difficult to tell if it will be well suited to particular political text analysis problems.

In the first half of this paper, I introduce the Wordscores algorithm, describe the problem of interpreting estimated document scores and the available rescaling solutions, and argue that no existing rescaling will work by demonstrating several fundamental problems in the method. In the second half of the paper, I show how to understand Wordscores as making classical ideal point assumptions about the relationship between document scores and words. After formulating a statistical ideal point model for words and comparing it to

Author's note: I would like to thank Ken Benoit, Mik Laver, Cees van der Eijk, and Wijbrandt van Schuur for useful comments and discussion. The remaining errors are my own.

© The Author 2008. Published by Oxford University Press on behalf of the Society for Political Methodology.
All rights reserved. For Permissions, please email: journals.permissions@oxfordjournals.org

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

existing work in political text analysis, I show how the model's structure and parameterization can avoid the problems identified in the first half of the paper. To support the claim that Wordscores approximates an ideal point model, I show first that Wordscores partly realizes an iterative method for computing a correspondence analysis and second that the parameters computed by correspondence analysis correspond to the word and document score parameters of the ideal point model. Finally, I specify the conditions under which correspondence analysis is a reasonable approximation to the ideal point model and relate problems with Wordscores to violations of particular conditions.

2 Wordscores

Given R documents or "reference texts" with known positions or scores on a policy dimension, Wordscores attempts to estimate the scores of L out-of-sample documents, the "virgin texts." To do so the method first estimates scores for each word type occurring in the reference texts and then combines these wordscores into a score for each virgin document. It is important to distinguish the two parts: estimating wordscores and estimating document scores using wordscores because they are, at least in principle, independent parts of the method. There is usually a third and final part of the method that rescales virgin document score estimates, so they can be more easily compared with the reference text scores.

Although the Wordscores algorithm is not explicitly derived from any statistical model of word generation, many aspects of the method can support such interpretations. In particular, the methods for assigning scores to words and documents have a symmetric probabilistic interpretation.

2.1 *Estimating Scores for Documents*

Wordscores computes the estimated score for a document d, θ_d, as the average of the scores, π_w, of the words contained in it[1]. When V types of words appear in a collection of reference texts and there are W word tokens in d,

$$\hat{\theta}_d = \frac{1}{W}\sum_{w}^{W} \hat{\pi}_w \tag{1}$$

$$= \sum_{j}^{V} \hat{\pi}_j \hat{P}(w_j|d). \tag{2}$$

The weighting probability is estimated by the proportion of tokens of each word type in the reference documents

$$\hat{P}(w_j|d) = \frac{c(w_j \text{ in } d)}{c(d)}, \tag{3}$$

where $c(\cdot)$ is the word token counting function. Laver, Benoit, and Garry (LBG) suggest

$$\hat{\sigma}^2(\hat{\theta}) = \sum_{j}^{V} [\hat{\pi}_j - \hat{\theta}]^2 \hat{P}(w_j|d) \tag{4}$$

as an estimator for the variance of this document score estimate, although it ignores sampling variation in $\hat{P}$.

[1]In fact, LBG assume only that virgin document scores are predicted by the average of the scores of their words, but since there is nothing special about virgin documents except our ignorance of their scores, this assumption must apply to all documents or it is impossible to explain why document scoring works.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

Equation (1) reflects the assumption that each observed word token provides the same amount of information about the document's score. Equation (2) emphasizes the same point at the level of word types: frequency is assumed to be a direct reflection of a word type's importance in determining a document's score. From this development, it is natural to assume that the true document score is (hats removed) simply

$$\theta_d = \sum_j^V \pi_j P(w_j|d). \tag{5}$$

2.2 *Estimating Scores for Words*

Wordscores computes the score for word w, π_w, as an average of document scores, weighted by the posterior probability of each document given that w occurs within it. When there are R documents, the word can appear in

$$\hat{\pi}_w = \sum_r^R \theta_r \hat{P}(d_r|w). \tag{6}$$

Posterior probabilities are computed in the following way: The probability of seeing w given that we are reading document i is given by equation (3). Assume that the prior probabilities of each reference document are equal, so $P(d_i) = 1/R$. The posterior probability of reading d_i after having seen w is then estimated as

$$\begin{aligned}\hat{P}(d_i|w) &= \frac{\hat{P}(w|d_i)P(d_i)}{\sum_r^R \hat{P}(w|d_r)P(d_r)} \\ &= \frac{\hat{P}(w|d_i)}{\sum_r^R \hat{P}(w|d_r)} \\ &= \frac{c(w \text{ in } d_i)/c(d_i)}{\sum_r^R c(w \text{ in } d_r)/c(d_r)}.\end{aligned} \tag{7}$$

Note that under this interpretation, $P(d_i)$ is a *prior* probability, so it would be inappropriate to estimate it from data, for example as $c(d_i)/\sum_r^R c(d_r)$. This is not because of Bayesian scruples but because the words in each document are sampled *conditional* on policy position, and for political text we know that the decision about what score to express and therefore what words to generate is not random but strategic and not explicitly modeled.

Although LBG do not make use of information about sampling variation in wordscore estimates, the document scoring procedure suggests that

$$\hat{\sigma}^2(\hat{\pi}) = \sum_r^R \left[\hat{\theta}_r - \hat{\pi}\right]^2 \hat{P}(d_r|w) \tag{8}$$

might be a reasonable estimator.

This development suggests that equation (6) should be understood as an estimate of the true wordscore, defined as

$$\pi_w = \sum_r^R \theta_r P(d_r|w). \tag{9}$$

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

Table 1 Mapping between LBG's notation and the notation used in this paper

Paper	LBG
$\hat{P}(w\|d_r)$	F_{wr}
$\hat{P}(d_r\|w)$	P_{wr}
$\hat{\pi}_w$	S_{wd}
$\hat{\theta}_w$	S_{vd}

Table 1 connects LBG's notation to the notation used in this paper. This is necessarily a partial mapping because LBG do not provide a way to distinguish population values from their estimates. Note also that all references to the dimension being scored, for example the d in S_{vd}, have been suppressed in order to focus attention on the estimation process. In this paper, d is used instead to refer to documents.

3 Problems with Document Scores

In applications, estimated document scores invariably have a *much* smaller variance than reference document scores and are bunched around $\bar{\theta}$, the mean of the reference document scores. For example, in LBG's U.K. party manifestos data the sample variance of the known scores is approximately 500 times larger than the estimated scores. This makes it difficult to compare estimated document scores with reference scores (Laver et al. 2003; Martin and Vanberg 2007), although Benoit and Laver (2007) have countered that raw estimated scores are nevertheless interpretable relative to each other. In an attempt to make the scores of virgin documents interpretable on the same scale as reference texts, two methods have been proposed for rescaling virgin document score estimates.

3.1 *Rescaling Document Scores*

LBG (2003) transform $\hat{\theta}$ into the more interpretable $\tilde{\theta}$ according to

$$\tilde{\theta} = \left[\hat{\theta} - \theta_{\text{vir}}\right]T + \bar{\theta}_{\text{vir}},$$

where T is the ratio of standard deviations of the reference and virgin document scores and $\bar{\theta}_{\text{vir}}$ is the average of estimated virgin document scores. This rescales virgin document scores to have the same variance as the original reference scores. Consequently, it only applies when more than one virgin document is to be scored.

LBG's transformation reflects an implicit assumption that the distribution of estimated document scores has the correct mean but the incorrect variance. This is problematic because in applications, the virgin score predicted mean is invariably close to $\bar{\theta}$ regardless of which virgin scores are scored, a shrinkage effect that is analyzed in more detail below.

In applications where the mean and variance of document scores can be expected to be approximately constant, the LBG transformation is very natural. LBG's empirical examples are panels of party manifestos, where it may be reasonable to expect policy position variance across elections to be stable. It is nevertheless worth noting the substantive implications of their transformation. Joint rightward or leftward movement of a set of parties relative to their positions in the previous election will be hard to discern because the mean of the virgin score estimates will always be close to that of the reference scores while the variance is not affected. Likewise an expansion of party positions to more extreme locations or increasing polarization in a legislature will also be masked.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

Martin and Vanberg (2007) have suggested the alternative rescaling transformation

$$\tilde{\theta} = \left[\hat{\theta} - \hat{\theta}_a\right]T + \hat{\theta}_a.$$

Here, $T = \left[\hat{\theta}_a - \hat{\theta}_b\right] / \left[\theta_a - \theta_b\right]$, where a and b index the reference documents with the lowest and highest scores, respectively. This transformation ensures that $\tilde{\theta}_a = \theta_a$ and $\tilde{\theta}_b = \theta_b$ and is most natural when these two anchoring documents can be identified on substantive grounds.

The transformation of Martin and Vanberg (MV) is valuable because it focuses attention on the important question of consistency. However, their notion of consistency is both very strong—θ_a and θ_b will be recovered exactly—and limited because it holds only for two documents. It may be better understood as enforcing a limited form of unbiasedness: samples of any size will always yield two correct estimates although the bias in remaining document score estimates will be unknown.

Practically, both the LBG and the MV rescaling operations are linear in T, so the variance of the document score estimates is $\hat{\sigma}^2(\hat{\theta})T^2$ with a corresponding correction to standard errors. But which rescaling transformation *should* be used?

Both transformations are derived from reasonable and general principles and yet yield different results in applications. In particular, the results of LBG (2003) cannot be replicated using the MV transform. In the light of these difficulties, we might try to avoid a decision by following LBG's suggestion to interpret untransformed scores. But although estimating scores for the reference texts as if they were virgin texts does successfully put all documents in the same scale, their relative positions on this scale do not replicate LBG's original analysis either.

A more basic problem with these transformations is that they shift the sensitivity of Wordscores output to different documents rather than removing it. LBG's transformation is insensitive to the composition of the reference document set but makes an estimated virgin document score depend on the composition of the virgin document set via its sample standard deviation. In contrast, MV's transformation is indifferent to the composition of the virgin document set but sensitive to the choice of anchoring reference texts. To get more insight, it is necessary to look more closely at the component processes of word and document score estimation.

4 A Closer Look at Score Estimation

Wordscores consists of two separate processes: the estimation of document scores from wordscores and the estimation of wordscores from document word counts. Introducing rescaling transformations is an attempt to fix their joint output but will not remedy any of the more basic problems with the component processes described below.

4.1 *Problems Scoring Documents*

The method for combining wordscores—averaging—and the method for transforming them into more interpretable forms—rescaling—are both linear, so they could be combined into one process to replace the process of averaging the wordscores of a virgin document with something more complex. But transforming output according to any linear mapping ignores a fundamental issue noted above: averaging wordscores to estimate a document's score implies that each word adds the same amount of information about the document.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

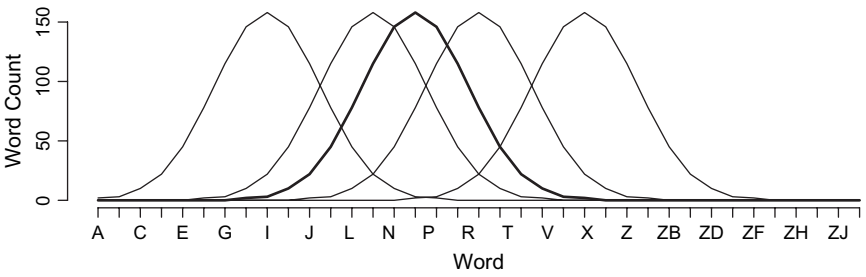


Fig. 1 Distribution of word counts for each document in the example data. These data are taken from Table 1 of Laver et al. (2003). Word count profiles for reference texts R1, R2, R3, R4, and R5 are plotted from left to right with the profile of virgin text V1 superimposed as a darker line.

In real text this is almost certainly false. Words like "taxes" are informative about economic policy in a way that words like "the" are not. However, Wordscores has no way to represent the difference between a genuinely informative politically centrist word—one that is used preferentially by center parties to denote centrist policy positions—and a word that all documents contain in roughly equal numbers for functional linguistic rather than political reasons. The problem is that if document scores are spread evenly across a policy dimension, then centrist words and politically uninformative words will both have wordscores close to $\bar{\theta}_{\text{ref}}$. Centrist words get centrist wordscores because $P(d|w)$ puts more weight on documents with scores near $\bar{\theta}_{\text{ref}}$ and uninformative words get centrist wordscores because indifference to policy position implies $P(d|w) \approx 1/R$, so the resulting wordscore is simply an average of the reference scores.

It is easy to see that the larger the number of word tokens with scores close to $\bar{\theta}$, the greater the movement of all estimated document scores toward $\bar{\theta}$. This effect is appropriate when all words are equally informative but overstated when there are also words with scores close to $\bar{\theta}$ simply because $P(d|w) \approx 1/R$. No linear transformation of document scores will fix this problem because it is the wordscore averaging process that is at fault. An ideal procedure would have to generate correspondingly extreme scores for just these informative words to offset the bias.

It is theoretically possible that a document could be scored as *more* extreme than any reference document, provided it is similarly constructed but contains a higher proportion of extremely scored words than any of the reference documents. However, in applications this effect will be swamped by the shrinkage due to uninformative words and by the effect of biases in wordscore estimation discussed next.

4.2 *Problems Scoring Words*

Problems relating to differing word frequency and informativeness will cause some of the shrinkage toward $\bar{\theta}$ that makes document score estimates so hard to interpret. But there are also problems with the method of assigning scores to words. These are most easily demonstrated using LBG's example data.

In LBG's example, there are $V = 37$ word types available, spread across 6 pseudo-documents containing 1000 word tokens each. These "words," shown in Fig. 1, are the 26 letters of the alphabet and the first 11 letters of the alphabet prefixed by the letter[2] Z.

[2]This is not quite the same as the paper, but since the words are arbitrary nothing depends upon it.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

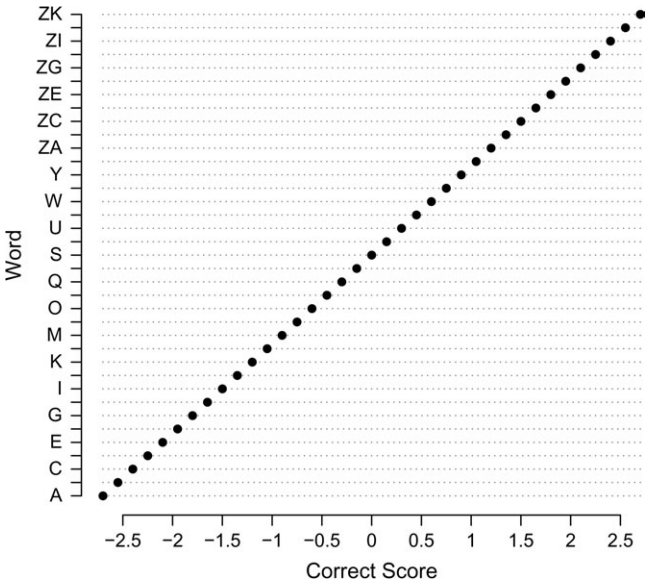


Fig. 2 Correct scores for words in the example data.

The reference documents R1, R2, R3, R4, and R5 are assigned scores of $-1.5, -0.75, 0, 0.75$, and 1.5, respectively. The task is to estimate the score of virgin text V1, whose word count distribution over the vocabulary is shown as a dark line in the figure.

From the word frequencies and document scores in the example, it is straightforward to identify a set of wordscores that fit the data perfectly: that is, scores that not only assign $\hat{\theta}_{V1} = -0.45$ without transformation but also score the reference documents correctly. Such scores are consistent in MV's sense, except that consistency holds over the complete document set. The scores, shown in Fig. 2, start at -2.7 for word "A" and increase in increments of 0.15 until they reach 2.7 for word "ZK."

In contrast, Fig. 3 shows scores estimated according to the standard Wordscores method. These agree with the correct scores where there is large amount of overlapping word frequency data but diverge at the edges where overlap decreases. In extreme cases, any word w that is unique to d has $\pi_w = \theta_d$ because $\hat{P}(d|w) = 1$ leading to strongly biased wordscore estimates. Unfortunately, such words are prevalent in real data. In LBG's 1992 U.K. party manifestos, approximately one half of all word types occurred in only one of the three manifestos.

In Fig. 3, the difference between the Wordscores estimates and correct scores appears only at the edges of the score distribution, but it is easy to show that this is not the only place it can arise. LBG's example uses a relatively large number of reference documents with scores that evenly span the range of possible document scores. Figure 4 shows the result of recomputing wordscores after removing the second and fourth reference documents. Here, only five estimated wordscores agree with the correct set of wordscores shown on the diagonal. The characteristic pattern of estimated scores in Fig. 4 will be familiar to users of the Wordscores because it appears in many data sets. For comparison, Fig. 5 shows the sorted economic dimension wordscores for the 1992 U.K. election manifestos. The vertical bands correspond to words that occur in only one manifesto.

There is nothing inherently wrong with assigning the same score to all the words that occur in only one document. This is because LBG define wordscores implicitly as any

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

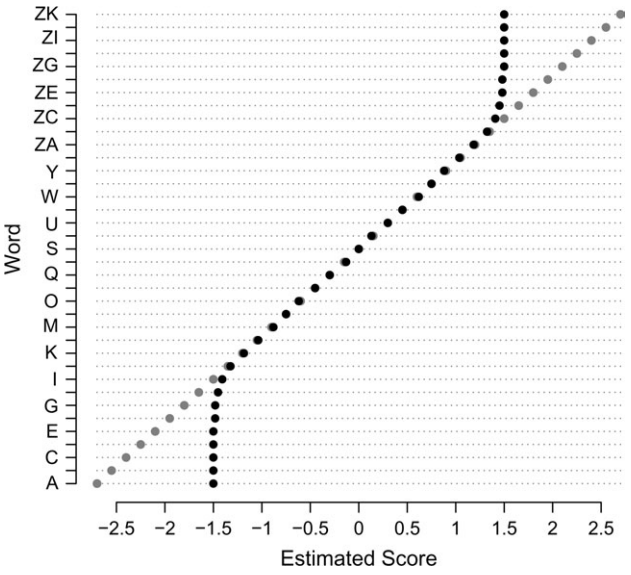


Fig. 3 Scores estimated according to the Wordscores method for the example data. Correct scores are marked in gray.

assignment of numbers to words that obeys the R sum constraints generated by equation (5), the reference document word counts, and their scores. The ''correct'' wordscores described above are therefore not unique because R constraints will not typically identify V wordscores[3]. Among the possible sets of correct wordscores, there will always be assignments where all words unique to a document are given the same score. The problem is that Wordscores will not in general assign these scores correctly. As an example, consider words A to E that are unique to document R1. Manual calculation shows that *any* assignment of wordscores that, when multiplied by the conditional probabilities of these words, generates the value -0.181 will obey the $R = 5$ sum constraints imposed by equation (5) and thus yield correct virgin and reference score estimates. So, if words A to E are to be assigned a single wordscore s, it should be the solution to

$$-0.18105 = 0.002s + 0.003s + 0.01s + 0.022s + 0.045s,$$

which is -2.208. This is not only quite different from the Wordscores estimate of -1.5 but also *larger* than any reference document score. It is therefore an example of the need to assign more extreme wordscores to offset shrinkage toward $\bar{\theta}$.

This example points to another fundamental problem with the Wordscores method of estimating scores for words and documents: equation (9) ensures that no wordscore can be *more* extreme than any of the document scores. However, if no wordscore can be more extreme than the lowest (or highest) document score, then that document score *cannot* be the average of the scored words within it as is required by equation (5). The method of generating wordscores is therefore incompatible with the method of scoring documents.

[3]For example, a trivial set of wordscores that estimates all documents correctly is to assign words I, N, P, S, X, and ZC the scores −9.49367, −4.746835, −2.848101, 0, 4.746835, and −9.49367, respectively (the scores for documents R1, R2, V1, R3, R4, and R5 multiplied by each word's conditional probability 0.158) and all other words the score 0.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

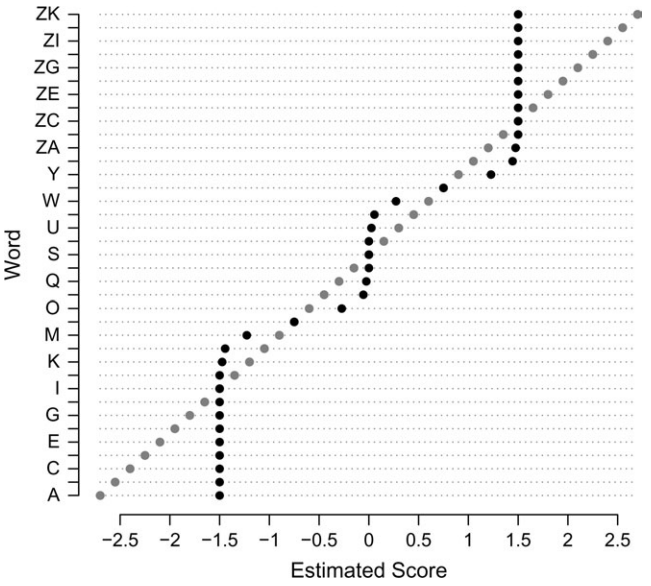


Fig. 4 Wordscores computed without reference documents R2 and R4 with scores −0.75 and 0.75.

In the example, the scores of the words appearing in the virgin document are well estimated because they are well within the range of the reference scores, so the virgin document score is itself well estimated. But where overlap is weaker, wordscore estimates can be strikingly biased. This is particularly problematic in applications where there are often

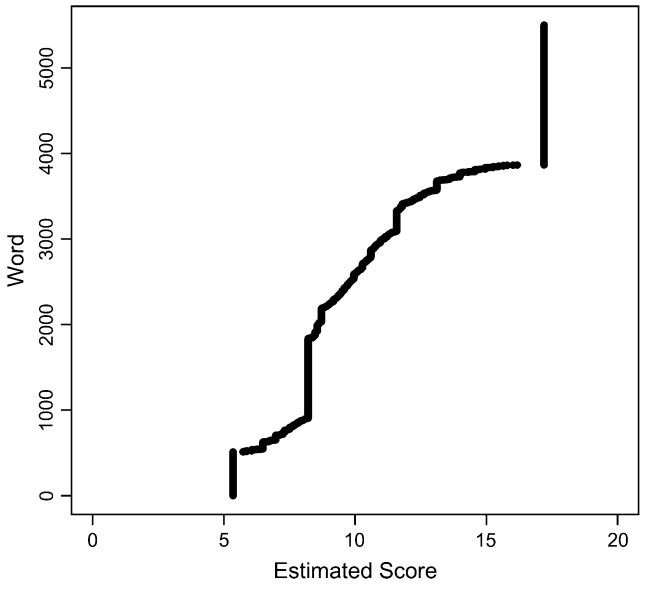


Fig. 5 Wordscores computed for the 1992 U.K. Labour, Liberal Democrat, and Conservative party manifestos on the economic dimension. Vertical bands of words with identical scores appear at the reference document scores 5.35, 8.21, and 17.2, respectively. Compare with Fig. 4.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

only *two* available reference texts, chosen to have maximally different policy positions, a choice that minimizes word overlap and risks biased wordscore estimates.

In summary, there are several fundamental problems with Wordscores' method of estimating scores for documents and words: first, the method has no way to distinguish the effects of word frequency and informativeness causing estimates to shrink toward apparently centrist policy positions. Second, it generates systematically biased wordscore estimates when there is insufficient overlap of word distributions across reference documents. Third, the assumptions of document scoring method are incompatible with the assumptions of the wordscore estimation method. That these problems can be demonstrated in example data that contain no error suggests they are basic to the method.

Despite these problems in the method, Wordscores can work well in applications, so at least some of the assumptions built into Wordscores must be appropriate to political text. These assumptions are best made explicit in a statistical model of the word generation process.

5 A Probability Model for Wordscores

The basic problem understanding Wordscores is that it appears to make no assumptions about the functional or distributional form of the mechanism by which words are generated from documents with particular scores. Wordscores estimates $P(w|d_i)$, irrespective of document scores, and then essentially asserts that the score of d_i on some policy dimension is θ_i. What is needed instead is an explicit form for $P(w|\theta)$, parameterized in a way that reflects Wordscores' assumptions about word generation and any *a priori* knowledge about the scores of particular documents.

A parametric form for $P(w|\theta)$ helps solve problems in both document and word scoring. For document scoring, adding explicit parameters representing how frequent and how informative a word is about policy position allows estimates of document scores to reflect the relative information available in each word count, rather than relying on averaging. Defining $P(w|\theta)$ also addresses the overlapping word count problem. Wordscores assigns words unique to a document the score of that document because $P(d_i|w) = 1$. However, with an explicit and reasonably smooth functional form for $P(w|\theta)$ the posterior distribution need not collapse to 1 over a single reference document score. Nearby possible scores will also affect the posterior via $P(w|\theta + \delta)$, despite the fact that no documents with a score of $\theta + \delta$ exist in the reference document set. Wordscores has no functional form to lean on between observed data points, so $P(w|\theta + \delta)$ is undefined.

5.1 *A Functional Form for Wordscores*

The word count distributions shown in Fig. 1 suggest that $P(w|\theta)$ should be a unimodal function centered on the document score. The Wordscores algorithm also supports this interpretation: Wordscores consists of two symmetrical weighted averaging procedures, the first to estimate scores for words and the second to estimate scores for documents. This averaging process has the effect of moving each π_w toward the center of the distribution of θ for the documents that contain w and to move each θ_d to the center of the distribution of π for the words in d. This process makes sense if Wordscores is in fact a classical ideal point model for words (Enelow and Hinich 1984).

With this intuition, it is possible to formulate a generalized linear latent variable model of the relationship between policy position and word generation (Bartholomew 1984; Elff 2008) in the same statistical framework as roll call voting analysis (Jackman 2001; Baker

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

and Kim 2004; Clinton et al. 2004). One simple model is that word counts are Poisson distributed with mean

$$\log \mathrm{E}[w] = c_w - \frac{1(\pi_w - \theta)^2}{2 \quad \tau_w^2}, \tag{10}$$

where c_w is the maximum probability of seeing word w and $\tau_w > 0$ is an informativeness term representing the rate of decrease in the word probability as π_w moves away from θ. This function has a maximum when $\pi_w = \theta$. In applications, all the models in this section will also require an offset to control for varying document length, suppressed here for clarity.

Intuitively, words with large c_w occur more frequently than words with small c_w in all documents, regardless of their policy position. Words with small τ_w are specific to a region of policy space around π_w and tend to appear only when documents express positions in this region. In contrast, the probability of seeing words with large τ_w does not depend strongly on the policy position of the document that contains them. τ_w therefore distinguishes between words that are frequent in all documents for functional linguistic reasons and those that are only frequent in documents expressing a centrist policy position. Both may have values of π_w toward the center and large c_w, but the former will have large and the latter small values of τ_w.

It is helpful to compare equation (10) with the models put forward by Monroe and Maeda (2004) and Slapin and Proksch (2008). In the simplest of these

$$\log \mathrm{E}[w] = c_w - \beta_w \theta, \tag{11}$$

where c_w represents the word probability when $\theta = 0$ and β_w represents the sensitivity of w's occurrence probability to changes in document policy position. Words that occur often in documents of all positions will have large c_w but small β_w. Like τ in equation (10), β_w distinguishes informative from uninformative words.

In contrast to the previous model, equation (11) has no parameters that can be identified as wordscores; words do not have a preferred position (π_w) in policy space, only a sensitivity (β_w) to changes in document policy position. Consequently, it cannot represent words that are used primarily to express centrist policy positions. Whether this is problematic depends on whether there are such words in political language, an empirical question that can only be answered by model comparison using real data.

Equations (10) and (11) are special cases of a quadratic model

$$\log \mathrm{E}\big[w\big] = b_0 + b_1\theta + b_2\theta^2. \tag{12}$$

To translate back to equation (10), let $\pi_w = b_1/2b_2$, $\tau = 1\big/\sqrt{-2b_2}$, and $b_2 < 0$ (to ensure a peak at $\theta = \pi_w$). To recover equation (11), let $b_2 = 0$ (ter Braak and Looman 1986).

Equation (10) offers the possibility of correcting at least some of the problems of word frequency, informativeness, and limited word overlap uncovered above. But what reason is there to think of Wordscores in ideal point terms? The next sections show how Wordscores is related to the ideal point model in equation (10) via the method of correspondence analysis.

6 Wordscores as Correspondence Analysis

Correspondence analysis (Greenacre 1993) is a method for extracting latent variables from a contingency table that has often been applied to linguistic data (Benzécri 1992). Given

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

a $V \times R$ matrix $\mathbf{C}$ where $\mathbf{C}_{wd}$ = c(w in d), a one-dimensional correspondence analysis associates a number θ with each column and a number π with each row such that the two sets of numbers have maximal correlation[4]. Correspondence analysis is usually presented as the result of an eigen decomposition, but for the purposes of understanding Wordscores it is more useful to note that the same solutions can be found using a simple iterative algorithm known as weighted or reciprocal averaging[5] (Hill 1973, 1974).

6.1 *Reciprocal Averaging*

The reciprocal averaging algorithm for a one-dimensional correspondence analysis starts by randomly choosing $\hat{\theta}_1, \ldots, \hat{\theta}_R$. It then computes

$$\hat{\pi}_w = \frac{\sum_d^R \mathbf{C}_{wd}\hat{\theta}_d}{\sum_d^R \mathbf{C}_{wd}} \tag{13}$$

for each row,

$$\hat{\theta}_d = \frac{\sum_w^V \mathbf{C}_{wd}\hat{\pi}_w}{\sum_w^V \mathbf{C}_{wd}} \tag{14}$$

for each column, and then normalizes the estimates of θ. Normalization is necessary to prevent repeatedly averaged quantities converging to a single value and is typically implemented by fixing the mean and variance of $\hat{\theta}_1, \ldots, \hat{\theta}_R$ at each iteration. Equations (13) and (14) and the normalization step are repeated until parameter changes are sufficiently small (ter Braak and Prentice 2004).

If columns of $\mathbf{C}$ contain compositional data, then $\sum_w^V \mathbf{C}_{wd} = 1$ and equation (14) reduces to

$$\hat{\theta}_d = \sum_w^V \mathbf{C}_{wd}\hat{\pi}_d. \tag{15}$$

To see the connection to Wordscores, note that the normalized word counts in equation (3) are compositional data of this type. Equation (15) is therefore identical to the document score estimator in equation (3) and equation (13) is identical to the wordscore estimator in equation (6), with conditional probability estimated as in equation (7). The final normalization step is performed in the Wordscores algorithm either by fixing R reference document scores and the (virgin) document score variance (the LBG method) or by anchoring with two documents (the MV method). Either is sufficient to identify the model and to prevent a degenerate solution.

Wordscores does not realize exactly this algorithm: initial document scores are not chosen randomly, the virgin texts are treated as out of sample, and Wordscores performs each step of the reciprocal averaging algorithm only once. But these are minor differences compared to the similarities in symmetrical structure. Wordscores is thus a single-step approximation to the reciprocal averaging algorithm for correspondence analysis.

[4]Many other criteria also lead to correspondence analysis as a matrix decomposition. Beh (2004) provides an extensive review.

[5]I would like to thank Wijbrandt van Schuur for suggesting this connection.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

Interpreting Wordscores as correspondence analysis connects the method to a well-developed statistical literature and also opens the possibility of extracting more than one latent dimension from data. However, it does not by itself help us understand what assumptions Wordscores makes about word generation since it is defined either algorithmically or in terms of a matrix decomposition rather than as probability model. However, correspondence analysis can be shown to be closely related to the ideal point model in equation (10).

7 Correspondence Analysis as Ideal Point Estimation

The relationship between correspondence analysis and ideal point estimation can be illuminated by looking at the maximum likelihood equations of θ_d and π_w in equation (10). These are (ter Braak 1985)

$$\theta_d = \sum_w^V \frac{\mathbf{C}_{wd}\pi_w}{\tau_w^2} \Big/ \sum_w^V \frac{\mathbf{C}_{wd}}{\tau_w^2} \; - \left[\sum_w^V \frac{(\theta_d - \pi_w)\mathrm{E}[w]}{\tau_w^2} \Big/ \sum_w^V \frac{\mathbf{C}_{wd}}{\tau_w^2} \right] \tag{16}$$

$$\pi_w = \frac{\sum_d^R \mathbf{C}_{wd}\theta_d}{\sum_d^R \mathbf{C}_{wd}} \; - \left[\frac{\sum_d^R (\theta_d - \pi_w)\mathrm{E}[w]}{\sum_d^R \mathbf{C}_{wd}} \right]. \tag{17}$$

If all words are equally informative, then τ cancels and the first terms on the right hand sides of equations (16) and (17) are the correspondence analysis recursions in equations (13) and (14).

Even if τ is shared, the terms in square brackets distinguish these equations from the correspondence analysis recursions. These terms will be small under two circumstances: when word probabilities are small and when word probabilities decrease in proportion to the distance between θ and π, that is, when words are generated according to the ideal point model in equation (10). The correspondence analysis recursions can be therefore be seen as approximations to the maximum likelihood equations for the parameters of an ideal point model for words.

To summarize the argument: Wordscores approximates a correspondence analysis because it performs a single iteration of the reciprocal averaging algorithm. And the reciprocal averaging algorithm for correspondence analysis approximates the maximum likelihood equations for an ideal point model because it ignores the bracketed terms in equations (16) and (17) and assumes a constant τ. How good are these approximations?

7.1 *Inconsistency and Bias*

In general, correspondence analysis estimators of word and document scores will be inconsistent. In the maximum likelihood estimation context, ideal point analysis practitioners are familiar with the ''incidental parameter problem'' that prevents consistent estimation of equation (10) or (11) for latent fixed effects θ when V is fixed and $R \to \infty$. Lynn and McCulloch (2000) prove the stronger result that π cannot be consistently estimated by correspondence analysis estimators in the same limit whether θ is treated as fixed or random.

Despite these inconsistency results, correspondence analysis and therefore Wordscores estimates of θ and π can under certain conditions correlate highly with their true values in finite samples (Lynn and McCulloch 2000). ter Braak provides a useful summary of the conditions under which bias (the bracketed terms in equations (16) and (17)) is minimized:

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

1. θ are equally spaced and extend over the whole range of π.
2. θ are closely spaced relative to τ.
3. π are equally spaced and extend past each θ in both directions.
4. π are closely spaced relative to τ.
5. τ_w is the same for all words.
6. c_w is the same for all words.

Assuming that equation (10) holds, these conditions describe the kind of text analysis problems in which Wordscores can be expected to work well. Conversely, violations correspond to the problems identified in previous sections and summarized below.

Wordscores treats all words as equally informative, providing no way to distinguish politically uninformative from centrist words or discount words that occur more frequently than others for linguistic rather than political reasons. Conditions 5 and 6 will therefore *never* hold for word count data because text exhibits highly skewed word frequency distributions regardless of genre (Zipf 1949; Mandelbrot 1954) and inevitably contains many uninformative words. Indeed, Fig. 2 of Slapin and Proksch (2008) supports the intuition that a word's utility in distinguishing policy position is not only quite variable but also inversely correlated with its frequency.

The second problem is that Wordscores generates biased wordscore estimates when there is insufficient overlap of word distributions between reference documents, as required by conditions 1 and 2. Conditions 3 and 4 remind us of a symmetrical problem in the document score estimation process when wordscores are inappropriately distributed. However, since words are more plentiful than documents this aspect of the insufficient overlap problem has less practical importance.

The third problem is that the assumptions of document and wordscore estimation used in Wordscores are incompatible. There would be no incompatibility if conditions 1 and 3 could hold simultaneously. However, that is impossible for any finite data set. Bias in wordscores, document scores, or both is therefore inevitable if correspondence analysis or Wordscores is used as an estimator.

There are typically many word types in a speech or political manifesto, so we might hope that they may relatively evenly spread out across a policy dimension. Then conditions 3 and 4 may be plausible. When many documents with known scores are also available, for example the speeches of a large number of legislators with varying and evenly distributed policy positions, conditions 1 and 2 might also be well approximated. Word and document scores should then be well estimated, except for those at the edges of the policy space because of the incompatibility problem. This is perhaps the best class of political text analysis problems for Wordscores. When there are very few documents with known scores, for example when analyzing party manifestos, then condition 2 will not hold even approximately. Large bias will therefore appear in wordscore estimates that will compromise new document score estimates.

If the parameters of equation (10) are estimated directly, for example by maximum likelihood or inferred using Bayesian methods rather than via the correspondence analysis or Wordscores approximations, then these biases should disappear. The empirical validation of this assertion is future work.

8 Conclusion

I have argued that Wordscores algorithm's computational straightforwardness, apparent absence of functional or distributional assumptions, and empirical effectiveness hide a number

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

of fundamental problems that are not solved by the rescaling transformations suggested in the literature. In order to address these problems, it is necessary to *understand* Wordscores. Understanding Wordscores involves determining what the method implicitly assumes about political text, particularly about the relationship between document policy positions and words. I have argued that Wordscores reflects an ideal point model for words and justified the claim in two steps. First, by showing that Wordscores is a partial implementation of the reciprocal averaging algorithm for correspondence analysis, and second that there is a close relationship between correspondence analysis and maximum likelihood estimation of some ideal point model parameters. To the extent that words in political text are generated according to an ideal point structure such as equation (10) rather than, for example, a factor structure like equation (11), Wordscores should be an effective method of inferring policy positions from documents. The empirical success of the method suggests that these assumptions may be reasonable. Conversely, I argued that although correspondence analysis, and therefore also Wordscores, will in general be inconsistent as an ideal point estimator, the nature and extent of its approximation to the ideal point model will determine the degree to which it will be biased in applications. In order to determine when Wordscores should work well, I list the conditions under which these biases can be expected to be small.

Funding

Enterprise Ireland (PC/2003/147).

References

Baker, F., and S. H. Kim. 2004. *Item response theory.* 2nd ed. New York: Wiley.

Bartholomew, D. J. 1984. *Latent variable models and factor analysis. Vol. 40*. London: Charles Griffin and Company Limited.

Beh, E. J. 2004. Simple correspondence analysis: A bibliographic review. *International Statistical Review* 72:257–84.

Benoit, K., and M. Laver. 2003. Estimating Irish party positions using computer wordscoring: The 2002 elections. *Irish Political Studies* 17:97–107.

Benoit, K., and M. Laver. 2008. Compared to what? A comment on ''A robust transformation procedure for interpreting political text'' by Martin and Vanberg. *Political Analysis* 16:101–11.

Benzécri, J.-P. 1992. *Correspondence analysis handbook*. New York: Marcel Dekker.

Clinton, J., S. Jackman, and D. Rivers. 2004. The statistical analysis of roll call voting: A unified approach. *American Journal of Political Science* 98:355–70.

Elff, M. 2008. *A spatial model of electoral platforms.* Annual meeting of the Political Methodology Society, Ann Arbor, Michigan.

Enelow, J. M., and M. J. Hinich. 1984. *The spatial theory of voting: An introduction*. New York: Cambridge University Press.

Greenacre, M. J. 1993. *Correspondence analysis in practice*. London: Academic Press.

Hill, M. O. 1973. Reciprocal averaging: An eigenvector method of ordination. *Journal of Ecology* 61:237–51.

Hill, M. O. 1974. Correspondence analysis: A neglected multivariate method. *Applied Statistics* 23:340–54.

Jackman, S. 2001. Multidimensional analysis of roll call data via Bayesian simulation: Identification, estimation, inference and model checking. *Political Analysis* 9:227–41.

Klemmensen, R., S. B. Hobolt, and M. E. Hansen. 2007. Estimating policy positions using political texts: An evaluation of the wordscores approach. *Electoral Studies* 26:746–55.

Laver, M., K. Benoit, and J. Garry. 2003. Extracting policy positions from political texts using words as data. *American Political Science Review* 97:311–31.

Lynn, H. S., and C. E. McCulloch. 2000. Using principal component analysis and correspondence analysis for estimation in latent variable models. *Journal of the American Statistical Association* 95:561–72.

Mandelbrot, B. 1954. Structure formelle des textes et communication. *Word* 10:1–27.

Monroe, B. L., and K. Maeda. 2004. *Talk's cheap: Text-based estimation of rhetorical ideal points.* Annual meeting of the Political Methodology Society. Stanford, CA.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press

Monroe, B., and K. Maeda. 2004. *Talk's cheap: Text-based estimation of rhetorical ideal-points.* POLMETH Working Paper.

Slapin, J. B., and S.-O. Proksch. 2008. A scaling model for estimating time-series party positions from texts. *American Journal of Political Science* 52:705–22.

ter Braak, C., and I. C. Prentice. 2004. A theory of gradient analysis. *Advances in Ecological Research: Classic Papers* 34:235–82.

ter Braak, C. J. F. 1985. Correspondence analysis of incidence and abundance data: Properties in terms of a unimodal response model. *Biometrics* 41:859–73.

ter Braak, C. J. F., and C. W. N. Looman. 1986. Weighted averaging, logistic regression and the Gaussian response model. *Plant Ecology* 65:3–11.

Zipf, G. K. 1949. *Human behavior and the principal of least effort*. Reading, MA: Addison Wesley.

https://doi.org/10.1093/pan/mpn004 Published online by Cambridge University Press