# Topic Modelling of British Parliamentary Debates

*Xiaoyu Sun* [1], *Zhi Rong Tan* [2]

[1 2] University of Chicago

xiaoyusun@uchicago.edu, tanzhirong@uchicago.edu

## Abstract

We performed topic modelling of the British Parliamentary debates in 2 eras, 1990s and 2010s, using 3 methods – LSA, LDA, and HDP. LDA works best for our given dataset. We discovered that while the focus on welfare and social policies has increased since the 1990s, that of economy and administration has decreased in the present decade. Challenges faced include varying word choices for a common topic in 2 different eras, and different integration of public issues within a topic across eras.

**Index Terms**: Topic Modelling, Latent Semantic Analysis, Latent Dirichlet Allocation, Hierarchical Dirichlet Process

## 1. Introduction

Transcripts of British parliamentary debates over the past 100 years offers a perspective on the evolution of public issues in the United Kingdom. Given that UK has evolved from a glorious imperial power into an aloof member struggling with Brexit of a regional union, we seek to understand how the national priorities have changed over the years, through topic modelling.

Specifically, we work on the speeches from 2 separate decades: 1990-1999 and 2009-2018 (referred as era 1 and era 2 respectively). These two eras are relatively close in time, so their topics would not be drastically different to allow for meaningful comparisons. Other objectives include exploring different topic modelling methods to find the best model, before analyzing both eras through changes in their topic types, word choices, and the proportion of documents each topic takes up.

## 2. Dataset

The original transcripts are sorted by dates, with 1 document per day on which parliament was convened. Using ElementTree in Python, we extracted all the major headings and paragraphs of speech under them. We discard short paragraphs with fewer than 400 characters, such as those entitled 'Preamble'. The output are 13,206 documents for era1 and 9,231 for era 2.

### 2.1. Tokenization & Unique Words Matrix

Data is preprocessed using the gensim and nltk libraries [1]:

1) Tokenization: Split text into lowercase words, and remove all punctuation.
2) Removal of stopwords (e.g. 'the', 'a', 'in'), and words with fewer than 3 characters
3) Lemmatization: Convert words to first person, present tense
4) Stemming: Reduce words to root forms

We create a dictionary containing every unique word from all the preprocessed documents, and removed tokens appearing in fewer than 5 documents or more than 50% of all the documents, because such words cannot provide much information concerning topics of the texts. The unique words form a dictionary for each era.

### 2.2. Tf-idf Matrix

Next, we extract the tf-idf values for each document. This numerical statistic reveals a word's importance to a document in the corpus. The simplest choice of term frequency for word $t$ in document $d$ is given by the frequency, i.e. $tf(t,d) = freq_{t,d}$, as indicated by its name. Inverse document frequency is a measure of how much information a word provides, i.e. whether it is common or rare across all document. It is defined as

$$idf(t,D) = log\frac{|D|}{|d \in D: t \in d|}$$

where $|D|$ is the total number of documents in the corpus $D$, and $|d \in D: t \in d|$ is the number of documents satisfying $tf(t,d) > 0$. Then tf-idf is calculated as $tfidf(t,d,D) = tf(t,d) \times idf(t,D)$. We collect the results for era 1 in a matrix of dimension (13206, 26456), and perform similar operations on era 2 documents.

## 3. Topic Modelling Techniques

We explore a few topic model methods – Latent Semantic Analysis, Latent Dirichlet Allocation, and Hierarchical Dirichlet Process. All 3 are bag-of-words models, in which text is represented as a multiset of words, disregarding grammar and word order but keeping multiplicity.

For each era, we split the dataset into the development and training set, in a 3:1 ratio. For tuning, we train the model using the training set, before finding the optimal hyperparameter with the highest coherence score from the development set. Finally, we will train the model on both the training set and development set, and find the final coherence score.

### 3.1. Intrinsic Coherence Score Measure

We evaluate each model using the average coherence score of the topic, where the optimal model/hyperparameter maximizes the average coherence score. [2] For a defined topic *V*, coherence is the sum of pairwise distributional similarity score over the set of topic word *V*, where *v* are the weighted words in *V*.

$$coherence(V) = \sum_{(v_i,v_j)\in V} score(v_i, v_j)$$

$$coherence(model) = \frac{1}{T}\sum_{t} coherence(V_t)$$

where $t=1, \ldots, T$, and $T$ = total number of topics.

More specifically, we use the UCI metric [3], which defines each word pair's score as the pointwise mutual information between the 2 words, taken over the sum of all words in the topic.

$$score(v_i, v_j) = log\frac{p(v_i,\ v_j)}{p(v_i)p(v_j)}$$

The final score of a model is calculated as the average of all topics' coherence scores, applied to the validation set's corpus. The word probabilities are computed by counting word co-occurrence frequencies in a sliding window over the validation set corpus.

We use UCI as it is an extrinsic metric calculated over an external corpus. There exist intrinsic metrics like UMass, where $p(v_i,\ v_j)$ is calculated based on the number of documents containing word *i* and *j*, computed over the original training corpus.

## 3.2. Latent Semantic Analysis

LSA is a linear dimensionality reduction method which performs Singular Value Decomposition on the matrix with the tf–idf value of each unique word in the dictionary, per document. [4] It assumes that words close in meaning occur in similar pieces of text. Recall: given $X \in \mathbb{R}^{D \times W}$, we can decompose the matrix by SVD into

$$X = UQV^T$$

where $U \in \mathbb{R}^{D \times D}$, $Q \in \mathbb{R}^{D \times W}, V \in \mathbb{R}^{W \times W}$. According to the Eckart-Young's Theorem, the best rank-r approximation is given by

$$X = U_r Q_r V_r^T$$

where $U_r \in \mathbb{R}^{D \times r}$ comprises of the first r columns of U, $Q_r \in \mathbb{R}^{r \times r}$comprises of the r largest singular values of X, and $V_r \in \mathbb{R}^{W \times r}$. This allows us to reduce the number of dimensions while preserving the similarity structure among columns.

Hence, *r* is the no. of topics, the topic-words matrix is given by $V_r^T \in \mathbb{R}^{r \times W}$, and the Document-Topic matrix is given by $U_r \in \mathbb{R}^{D \times r}$. The lower-dimension representation of a document $d_j$ in $\mathbb{R}^W$ is given by $V_r^T d_j \in \mathbb{R}^r$.

The only hyperparameter tuned is *r*, the number of topics.

### *3.2.1. Training*

For both eras, we proceed with training our LSA model on the training set at a fixed number of topics, before evaluating the coherence score. For Era 1, the optimal n = 7, with a final coherence score of -0.0829.

An issue with LSA is that the topic model is not easily interpretable. For instance, see example below:

| Topic | Words |
|---|---|
| **1** | -0.386*"amend" + -0.317*"claus" + -0.315*"insert" + -0.211*"page" + -0.195*"subsect" + -0.148*"ireland" + -0.143*"line" + -0.142*"section" + 0.133*"tax" + -0.131*"lord" |

For Era 2, the optimal n = 6, with a coherence score of -0.823.

| Topic | Words |
|---|---|
| **Finance** | 0.148*"tax" + 0.103*"amend" + 0.100*"polic" + 0.099*"school" + 0.097*"nhs" + 0.096*"bank" + 0.086*"budget" + 0.084*"vote" " + 0.083*"economi" |
| **Administrative** | 0.270*"amend" + 0.254*"forthwith" + 0.171*"commenc" + 0.170*"lord" + 0.168*"draft" + 0.163*"conclus" + 0.160*"claus" + 0.159*"conclud" |
| **Healthcare/ Education** | 0.091*"school" + 0.074*"request" + -0.065*"chancellor" + 0.064*"nhs" + 0.061*"urg" + -0.055*"bank" + 0.052*"hospit" + -0.051*"economi" + 0.050*"patient" |
| **Law** | -0.206*"prison" + 0.194*"proceed" + -0.194*"polic" + |
| **Enforcement** | 0.155*"petition" + 0.148*"petit" + 0.136*"bank" + 0.123*"chancellor" + -0.122*"crime" |
| **Brexit** | 0.169*"european" + -0.166*"ireland" + -0.164*"petit" + -0.161*"petition" + -0.156*"referendum" + -0.155*"elector" + -0.138*"brexit" + -0.136*"northern" |
| **(Unclear)** | -0.270*"approv" + -0.269*"lay" + -0.223*"amend" + -0.195*"forthwith" + 0.138*"conclud" + 0.133*"conclus" + 0.126*"commenc" + -0.125*"regul" + -0.116*"nhs" |

Compared to Era 1, Era 2's model is more easily interpretable, although there are still certain topics that are confusing.

In summary, LDA's advantages lie in its simplicity (as a linear model employing SVD) and speed. However, a concern is that the results may not be perfectly interpretable.

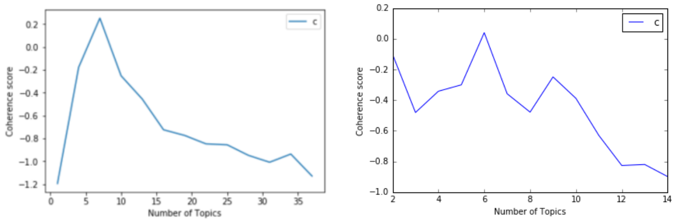

Figure 1: *Coherence Score for No. of Topics, Era 1 (Left) and Era 2 (Right)*

## 3.3. Hierarchical Dirichlet Process

We will next explore the Hierarchical Dirichlet Process [5], which is similar to LDA, but rather than fixing the number of topics, this number is generated by a Dirichlet process, and is random. A nonparametric Bayesian approach, the "hierarchical" portion refers to another layer being added to the generative model to produce the number of topics.

The method uses a Dirichlet process (DP) for each group/topic of data, with the DP for all groups sharing a base distribution which is itself drawn from a Dirichlet process.

$$G_0 \sim DP(\alpha_0, H) \qquad G_j \sim DP(\alpha, G_0) \text{ for each j}$$

The base distribution is a symmetric Dirichlet over the vocabulary simplex. Also note that $G_0 = \sum_k \pi_{0k} \delta_{\theta_k}$which implies the distribution $G_0$ has infinite support.

First, the model generates topic associated with the n-th word in the j-th document, then generate the word from the topic.

$$\theta_{jn} \sim G_j$$

$$w_{jn} \sim multi(\theta_{jn})$$

At the document-level draw, $G_j$ inherits the topic from $G_0$ but weights them according to document-based topic proportions. More information can be found from the Stick-Breaking Process, and Chinese Restaurant Process.

The hyperparameters tuned are $\alpha, H$.

### *3.3.1. Training*

Era 1: In the tuning of the hyperparameters, we proceed first with tuning the value of (scalar) $\alpha$, before proceeding to tune $H$. To reduce article length, only the tuning results for $\alpha$ is shown.

We selected a value of $\alpha = 0.1, H = 0.1$, and obtained a model with a large number of topics (~50), with a Coherence score of -9.9. What was interesting is that while the first few topics are interpretable (see topic 1 below), the next few becomes drastically challenging to interpret.

| Topic | Words |
|---|---|
| **1. Budget** | 0.001*tax + + 0.001*labour + 0.001*school + 0.001*claus + 0.001*pension + 0.001*unemploy + 0.001*invest |
| **…** | …. |
| **Topic 5** | 0.001*junctur + 0.001*unreport + 0.001*interbre + 0.001*brokerag + 0.000*horizont + 0.000*climact |
| **Topic 10** | 0.001*courthous + 0.001*nomenclatur + 0.000*benidorm + 0.000*wallenberg + 0.000*creepi + 0.000*room |

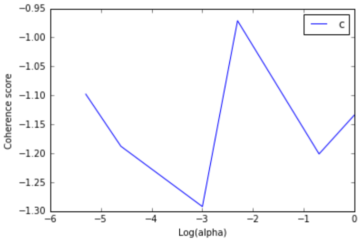

Figure 2: Tuning of $\alpha$

For Era 2, we obtained a value of $\alpha = 0.2, H = 0.15$, with a coherence score = -10.1. Interestingly, we encountered the same issue as Era 1 – the first few topics are easy to distinguish, but the next few became drastically challenging.

| Topic | Words |
|---|---|
| **1. Finance** | 0.000*bank + 0.000*cent + 0.000*amend + 0.000*scotland + 0.000*payment + 0.000*claus + 0.000*tax + 0.000*disabl |
| **…** | …. |
| **5.** | 0.000*courthous + 0.000*nomenclatur + 0.000*benidorm + 0.000*wallenberg + 0.000*creepi + 0.000*room + |
| **10.** | 0.000*skylin + 0.000*goug + 0.000*portland + 0.000*unfurnish + 0.000*achill + 0.000*charterhouse |

We hypothesize a few possibilities to the poor performance. One limitation of HDP as a nonparametric method is that the posterior inference algorithm requires multiple hyperparameters. Hence, in running the algorithm, we did not optimize many other parameters such as $\gamma, \kappa$, or the iteration and convergence time, instead opting to use the default option in the package – further tuning may have improved the final model performance.

## 3.4. Latent Dirichlet Allocation

LDA [6] is a generative model, where documents are viewed as mixtures of topics, and each topic $t$ has multinomial distribution with parameter $\beta_t$ over words, $\beta_t \in [0,1]^M$, $\sum_{m=1}^{M} \beta_{tm} = 1$, where $M$ is vocabulary size. In addition, each document $d$ has multinomial distribution with parameter $\theta_d$ over the topics. For a corpus $D$ where each document $d$ has length $N_d$, the LDA generative process is as follows.

1) For each $d$, generate $\theta_d \sim Dir(\alpha)$, where $\alpha$ is the parameter of the Dirichlet prior on document topic distribution

2) For $t = 1, \dots, T$ (number of topics), generate $\beta_t \sim Dir(\eta)$.

3) Fix document $d$. For each $n = 1, \dots, N_d$, draw topic $Z_n \in \{1, \dots, T\}$ from Multinomial($\theta_d$), and then draw word $W_n$ from Multinomial($\beta_{Z_n}$).

The hyperparameters to be tuned are $\alpha$ and $T$.

### *3.4.1. Training*

Era 1: we obtained a model of T=7, and $\alpha = 0.5$, with a coherence score of -0.0591.

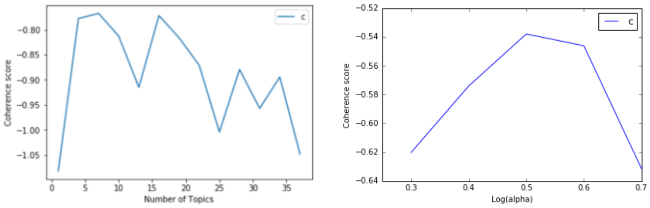

Figure 3: *(Left) Coherence Score for No. of Topics; (Right) Score for $\alpha$ , Era 1*

Our topic model is:

| Topic | Words |
|---|---|
| **Finance** | 0.006*"deposit" + 0.003*"fee" + 0.003*"deem" + 0.003*"refund" + 0.003*"chargeabl" + 0.002*"privat" + 0.002*"relief" + 0.002*"tax" + 0.002*"acquisit" |
| **Agriculture & Environment** | 0.006*"farmer" + 0.005*"beef" + 0.004*"signifi" + 0.004*"agricultur" + 0.004*"food" + 0.004*"farm" + 0.003*"environment" + 0.003*"queen" + 0.003*"fisheri" |
| **Healthcare & Social Policy** | 0.003*"health" + 0.002*"hospit" + 0.002*"prison" + 0.002*"pension" + 0.002*"nhs" + 0.002*"patient" + 0.002*"crime" |
| **Foreign Policy & Defence** | 0.002*"kosovo" + 0.002*"iraq" + 0.002*"saddam" + 0.002*"bosnia" + 0.002*"nato" + 0.001*"militari" + 0.001*"troop" + 0.001*"iraqi" + 0.001*"serb" |
| **Education & Welfare Policy** | 0.004*"tax" + 0.004*"school" + 0.003*"educ" + 0.003*"industri" + 0.002*"labour" + 0.002*"invest" + 0.002*"unemploy" + 0.002*"pension" |
| **Religion** | .013*"church" 0.004*"clergi" + 0.002*"resum" + 0.002*"bishop" + 0.002*"dioces" + 0.002*"synod" + 0.002*"parish" + 0.001*"cathedr" |
| **Administrative** | 0.006*"insert" + 0.006*"claus" + 0.004*"page" + 0.004*"motion" + 0.004*"andrew" + 0.004*"section" + 0.004*"paragraph" + 0.003*"schedul" + 0.003*"proceed" |

Era 2: We obtained a model of T=6, and $\alpha = 0.5$, with a coherence score of -0.4991. Our topic model is:

| Topic | Words |
|---|---|
| **Education & Healthcare** | 0.002*"school" + 0.001*"educ" + 0.001*"student" + 0.001*"pupil" + 0.001*"young" + 0.001*"nhs" + 0.001*"hospit" + 0.001*"nurs" |
| **Social & Welfare Policy** | 0.002*"prison" + 0.002*"tax" + 0.002*"women" + 0.001*"disabl" + 0.001*"pension" 0.001*"payment" + 0.001*"employ" + + 0.001*"authoris" |
| **Economy** | 0.002*"trade" + 0.002*"union" + 0.001*"tax" + 0.001*"invest" + 0.001*"rail" + 0.001*"industri" + 0.001*"bank" + 0.001*"sector" + 0.001*"economi" |
| **Brexit / Border Issues** | 0.005*"ireland" + 0.004*"northern" + 0.002*"brexit" + 0.002*"european" + 0.002*"petition" + 0.002*"border" + 0.002*"union" + 0.002*"leasehold" |
| **Administrative** | 0.005*"draft" + 0.004*"amend" + 0.004*"approv" + 0.004*"lay" + 0.003*"conclus" + 0.003*"commenc" + 0.003*"lord" + 0.003*"conclud" + 0.002*"deleg" |
| **Foreign Policy/ Defence** | 0.002*"russian" + 0.002*"nato" + 0.001*"arm" + 0.001*"defenc" + + 0.001*"saudi" + 0.001*"yemen" + 0.001*"syria" + 0.001*"weapon" + 0.001*"humanitarian" |

LDA gives us the clearest topic model across the 3 methods. Recall that LSA has the simplest training process, but produces unclear topics that is hard to interpret, while HDP has multiple parameters to consider when training; if they are not meticulously tuned, the topic models are the worst. Since LDA performs best in terms of interpretability and score, we shall proceed with the comparison using the LDA model for both eras.

# 4. Document Comparison

## 4.1. t-SNE Visualization

To better understand the separation of topics into different categories, we employed t-SNE dimensionality reduction to transform the document-topic matrix $X \in \mathbb{R}^{D \times T}$ into $Z \in \mathbb{R}^{D \times 2}$. Each point represents a document, and 2 documents have the same color if their *maximum-probability* topic is the same.

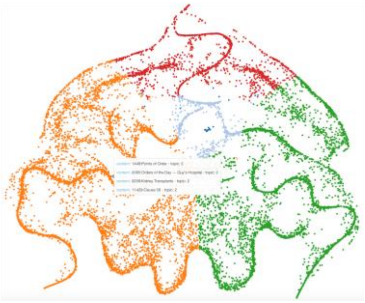

Figure 4: *t-SNE, Era 1*

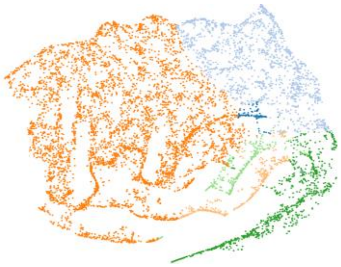

Figure 5: *t-SNE, Era 2*

## 4.2. Era Comparison

| Debate Focus | Era 1 | Era 2 |
|---|---|---|
| Administrative | 30% | 20% |
| Finance / Economy | 25% | 20% |
| Social Policy (Health, Education, Welfare) | 25% | 40% |
| Religion | ~1% | N/A |
| Foreign Policy / Defence | 10% | 10% |
| Agriculture / Environment | ~9% | N/A |
| Border Issues | N/A | ~10% |

Table 1: *(Approximate) Policy Focus across Eras*

From t-SNE, we roughly summarize the proportions of each major policy areas for the two eras in Table 1. If one topic has a clear cluster, we just estimate its proportion as the size of the cluster in the plot, e.g. the Agriculture topic represented by light blue in Figure 4 takes up ~ 10% of the whole dataset, verified by counting the number of documents $N$ that has agriculture topic with the largest probability. Then, proportion = *N/D.*

But if the cluster is large, we are wary of documents of different nature being labelled under a similar max-probability topic. For example, $\theta_A = [0.4, 0.39, 0.01], \theta_B = [1,0,0]$ are both labelled as topic 1 in the plot. However, observe they have a completely different topic probability structure. Depending on their position in the plot, and close examination of the headings, we may choose to consider document A as a separate debate focus from B. Acknowledging the subjectivity of this method, we round off our values to the nearest 5.

# 5. Discussion

## 5.1. Analysis

From 1990s to 2010s, a few changes are observed. First, notice that there has been a decrease in the proportion of time spent on administrative issues. In general, the policy focus on foreign policy and finance/economy has remained roughly equal, although the latter saw a slight drop in the 2010s. Most significantly, the proportion spent on social policy (hereby defined as the combination of healthcare, education, welfare and domestics) have increased from 25% to 40%.

Interestingly, while agriculture/environment takes up 10% in 1990s, this category disappeared, and is replaced with a new border issue category focusing primarily on Brexit issues and UK's position in the EU. However, a closer look reveals that agriculture is still heavily discussed in the 2010s. However, they are completely contained within the issue of Brexit. Further explanation is found here. [7]

## 5.2. Challenges

The optimal model for both eras produces roughly the same number of topics after tuning, at 7 and 6 respectively. However, observe that each topic is a combination of more policy issues – e.g. healthcare + education, foreign policy + defence. The topics could possibly be broken down into smaller topics, if we had picked T=12 or 16 (2nd highest coherence score). Yet, the small number of 6/7 reveals that policy issues are highly integrated – for example, finance is heavily related to the economy, while healthcare is related to welfare policy. What is initially surprising is that the "policy issue" integration could differ across eras – in 1990s, healthcare is combined with social policy, while education is combined with welfare. In 2010s, healthcare is now combined with education! While these different combinations of policy issues into topics cause comparisons to be challenging, they reveal insights into how correlations between policy issues change over eras.

We also explored combining both eras into a single dictionary, before running a single topic model on them. Yet, because the words are specific to each era (e.g. Brexit, Gulf War, Eurozone Crisis), combining them may just cause each era to fall into its own topics instead of a combined topic model.

## 5.3. Extensions

Further analysis of the British parliamentary debate dataset can be performed using other methods. Some extensions include:

1. Moving from unigram / bag-of-words model to bigram/trigram models using word embeddings
2. Consider deep learning methods – e.g. LDA2Vec employs Word2Vec to obtain vector representation in LDA

These methods should further enhance the information revealed in the topic models, and give rise to refreshing insights.

# 6. Acknowledgements

Xiaoyu is responsible for web data scraping and data processing, including the calculation of tf-idf matrix. Zhi Rong completes the project scope, research of the 3 topic modelling methods, and final analysis. Both assisted with the model training for the model modelling and visualization, each taking charge of 1 era.

## 7. References

[1] Li, Susan. (2018). Topic Modeling and Latent Dirichlet Allocation (LDA) in Python. *Towards Data Science*

[2] Newman, David et al. (2010). Automatic Evaluation of Topic Coherence. *Annual Conference of the North American Chapter of the ACL*, p100-108.

[3] Steven, K., et al. (2012). Exploring Topic Coherence over many models and many topics. *2012 Joint Conference on Empirical Methods in NLP*, p952-961.

[4] Landauer, T.K., et al. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes*, p259-284.

[5] Teh, Y.W., et al. (2005). Hierarchical Dirichlet Processes.

[6] Hofmann, T. (1999). Probabilistic latent semantic indexing. *22nd SIGIR conference*, p50-57.

[7] Irish Farmers' Association. (2017). Brexit: The Imperatives for Irish Farmers and the Agri-Food Sector. *IFA Policy Paper*.