# 

## Introduction

The aim of the report is to answer two questions: (1) to what extent do MPs tend to ask questions directly referencing their own constituency or a location within it; (2) what (if any) identifiable regional differences are there in the types of questions asked? The report starts by detailing data preprocessing conducted for both questions. Then the specific method for each question is divided into two.

## Required Modules

Table 1 shows the frameworks and libraries used to preprocess, model and visualise the data.

|  |  |  |
| --- | --- | --- |
| **Module Name** | **Question** | **Description** |
| SPARQLWraper | 1 | Interface for making request to SPARQL endpoints. |
| Stanza | 1 | Natural Language Processing (NLP) framework created by Stanford. |
| TheFuzz | 1 | Fuzzy string matching library. |
| WordCloud | 2 | Visualise word frequency with word clouds. |
| pyLDAvis | 2 | Interactive graphs for inter-topic distance measuring and the most relevant terms for a given topic model. |
| Gensim | 2 | Provides models such as LDA, as well as common text preprocessing functions for NLP. |
| Pandas | Both | Data analysis library. |
| MatplotLib | Both | Data visualisation library. |
| NLTK | Both | Suite of methods for NLP such as tokenization, stop-word removal, etc. |

Table - Description of Modules for the Project

## General Data Preprocessing

Questions asked between the 1st January 2023 to the 30th September 2023 were collected from the UK Parliament’s API[8]. The data consisted of the form described in Table 2.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Required** | **Data Type** | **Description** |
| **question** | Yes | String | Question URI |
| qnum | Yes | Int | Question Number |
| **text** | Yes | String | Question Text |
| date | Yes | Date | Date question was asked |
| person | Yes | String | Asking person’s URI |
| **name** | Yes | String | Asking person’s first name |
| **surname** | Yes | String | Asking person’s surname name |
| seatIncumbency | No | String |  |
| seat | No | String |  |
| constituency | No | String |  |
| house | No | String |  |
| houseName | No | String |  |
| seatIncumbencyStartDate | No | Date |  |
| **constituencyName** | No | String |  |
| constituencyStartDate | No | Date |  |

Table - Description of Parliamentary API Data

|  |  |  |  |
| --- | --- | --- | --- |
| **Question Data** | Total Questions | Current MP’s Questions | Non-MP’s Questions |
| **Questions Count** | 29,552 | 25,145 | 4407 |

Table - Question Counts

In total there were 29,552 written questions asked in the given timeframe. Of the total, 4407 were filtered out because they did not have an associated MP.

The data was stripped to only include the fields outlined in Table 2 in bold. Figure 1 visualises the counts of the number of questions asked per constituency during the time period. It should be noted that MP’s can ask an unlimited number of written questions, however the dataset was biased towards some regions.

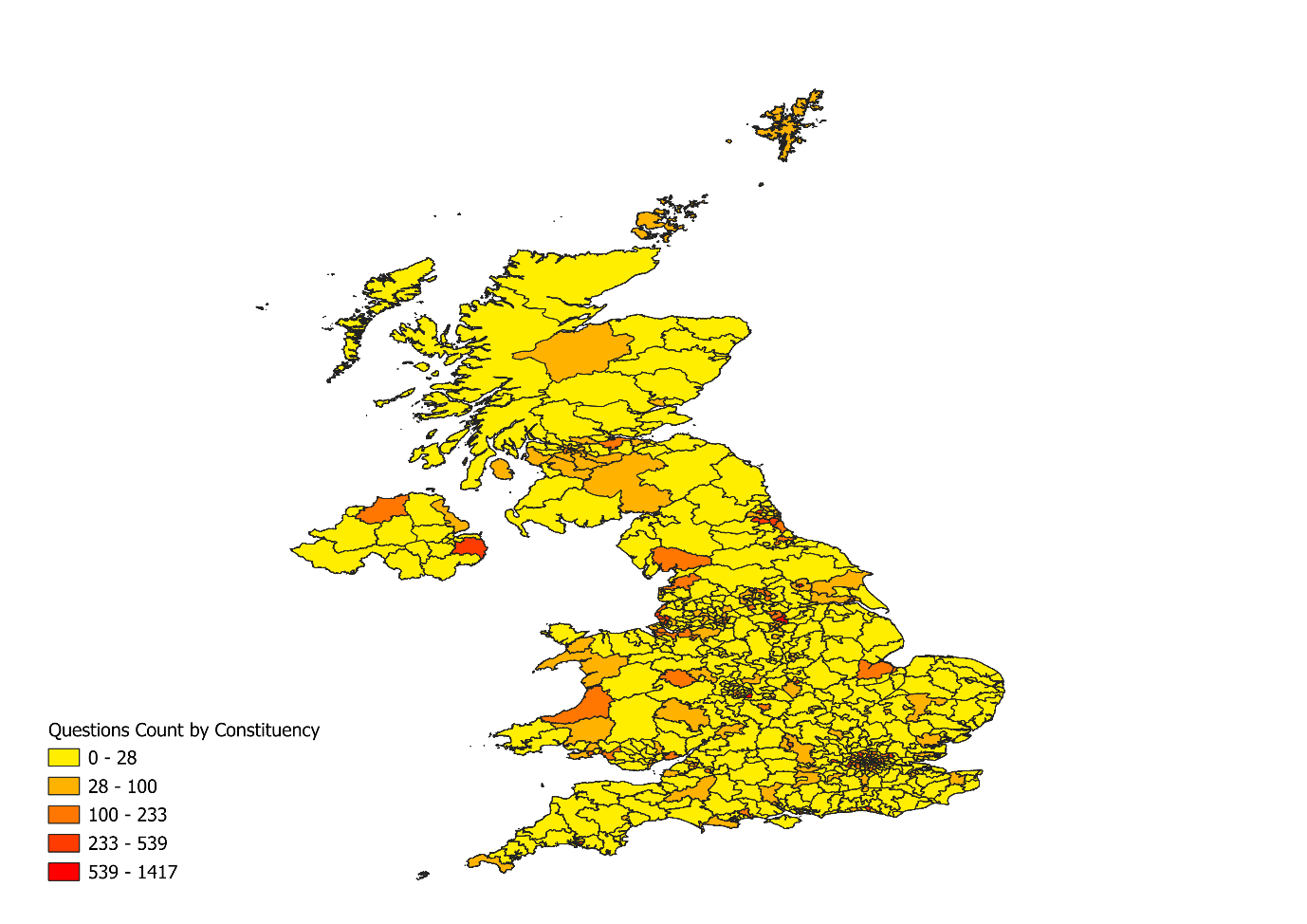
****

Figure - Question Count by Constituency

# To what extent do Members of Parliament (MPs) tend to ask questions that directly reference their own constituency or a location in it?

## Data Preprocessing

In addition to collecting MP’s questions, a collection of entities in the UK were gathered. Initially, 702,004 unique entities were gathered from WikiData[15]. The WikiData consisted of the fields as seen in Table 4.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Required** | **Data Type** | **Description** |
| place | Yes | String | Place URI |
| placeLabel | Yes | String | Place Label (English) |
| constituency | Yes | String | Constituency URI |
| constituencyLabel | Yes | String | Constituency Label (English) |
| coordinate | Yes | Point | Coordinate in WGS84 |
| instance | Yes | String | Entity Class URI |
| instanceLabel | Yes | String | Entity Class Label |

Table - Description of Places Query from WikiData

The query was constrained to entities with the UK as the associated country. To maximise the range of places gathered, constraints on the entities were kept to a minimum. This resulted in a moderate number of entities that were not place-like. As such, the coordinates associated with the places were also obtained for further cross-referencing against the ONS’s Parliamentary Electoral Constituency Boundaries[4]. Furthermore, obtaining the constituency associated with an entity was very unreliable. Instead of disregarding such data, the constituencyLabel for an entity was converted into a list of labels to capture valid instances of entities which spanned multiple constituencies. However, to ensure that each place contained a valid constituency the data was loaded into QGIS, where it was merged with the ONS’s electoral boundaries. This also eliminated instances which were not place-like or were located outside the UK. This reduced the total number of entities by 26,938, as seen in Table 5.

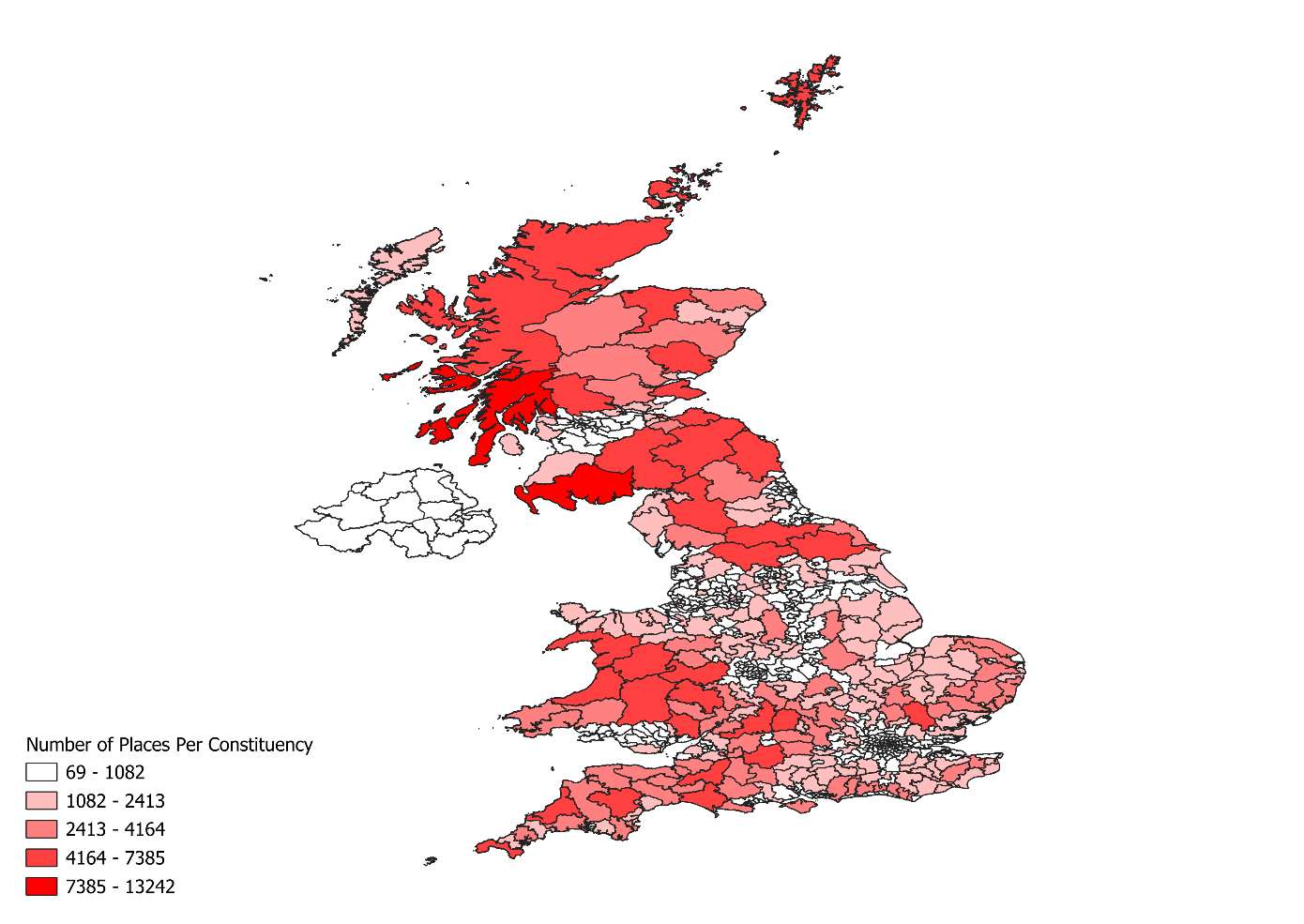
****

Figure - Place count by constituency

Figure 2 highlights the distribution of places by constituency. It should be noted that although smaller constituencies have less places, they tended to have a higher proportion for the given area.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data** | **WikiData** | **WikiData on Mainland** | **ONS Places** | **ONS Places with Constituency** | **Combined** |
| **Count** | 702,004 | 675,066 | 108,464 | 98,024 | 764,177 |
| **Instances** | 3112 | N/A | N/A | N/A | N/A |
| **Constituencies** | 413 | N/A | 633 | N/A | N/A |

Table - Overview of Place Data

In addition, the ONS’s Index of Place Name’s [7] was combined with the WikiData to improve the robustness of the dataset as seen in Table 4. Place names were converted to lowercase to avoid duplication in the dataset. It should be noted that the data will have contained some duplicates, however this would not have affected the performance of the model as each question could only have a reference or not.

Figure 3 highlights the top 10 most frequent instances of places from WikiData after the entities were constrained to within the UK by QGIS.

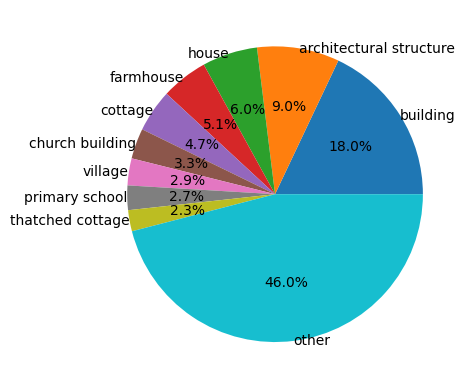


Figure - Distribution of Place Types

Additionally, the Parliamentary API was used to gather the name of the 650 electoral constituencies which was used for cross-referencing with WikiData. A separate query was used to obtain constituency and instance labels.

## Model

Modelling consisted of three components. The place entities first had to be identified. Then, the identified entities could be cross-referenced with the place data collected. Finally, the questions with references could be counted for each constituency.

### Named Entity Recognition

NER extracts the entities of interest. Instead of searching the entirety of each question for place matches, the shortlisted named entities can be searched instead. There are various libraries in Python that enable NER, including Stanza, Spacy, NLTK and Flair. Ideally, one would train a custom model for a given task. However, this requires pre-labelled data and as such a pre-trained model was used.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Framework | Model Type | Classes | Training Data | Performance |
| NLTK Stanford Java Named Entity Recogniser | CRF | 3 | CoNLL03 and more | Unknown |
| Stanza NER | Bi-LSTM Sequence Tagger with CRF-based decoder | 18 | Ontonotes | 88.8% |

Table - NER Models

Initially, the Natural Language ToolKit (NLTK) was used with its built-in interface with the Stanford NER Tagger[3]. Specifically, the three classes model was used which identifies location, persons and organizations. The NLTK implementation of the Stanford NER Tagger does not retain information of boundaries for named entities. As such the named entities were often divided into multiple words and the model was very inefficient. The Stanford team created the Stanza library to provide better support for their models[9]. For this paper the neural network pipeline was used for NER. This consists of a model trained on the Ontonotes dataset[14]. The model had 88.8% accuracy which was only marginally worse than the performance of the Flair module[9]. This consisted of 18 named entities, however the “Person” tag was excluded.

### Entity Geolocation

The second part of modelling was determining whether the tagged entities in the question belonged to the questioner’s constituency. Each tagged entity had to be checked against the place dataset.

For each tagged entity, the entity was compared with the questioner’s constituency and flagged accordingly. Then for each matching entity in the places dataset, their constituency was equated with the questioner’s constituency.

|  |  |
| --- | --- |
| Flag | Description |
| Is\_ref | If question contained a reference to the constituency name or a location within it. |
| Is\_direct\_ref | If question contained a direct reference to the constituency name. |
| Is\_fuzzy\_ref | Is\_ref where Levenshtein distance of 2 is acceptable when comparing constituency of place in question and place in the places dataset. |

Table - Reference Flag Description

Table 7 shows the flags given to questions by the model. As the constituency’s name in the places dataset were obtained from a different source it was possible that the model would have some false negatives. As such, the model was amended to have a 3rd flag, “is\_fuzzy\_ref”.

### Summary of Results

The data was accumulated into a separate DataFrame. Of the 25,145 questions only 1,086 of the questions contained a reference. Furthermore, 526 of the questions were a direct reference to the constituency’s name. Fuzzy matching only increased the number of references by 0.04% as seen in Table 8.

|  |  |  |  |
| --- | --- | --- | --- |
| **Type of Reference** | **Reference** | **Direct Reference** | **Fuzzy Reference** |
| **Count** | 1,086 | 526 | 1,096 |
| **Percentage of Total Questions** | 4.31% | 2.09% | 4.35% |

Table - Number of References by Type

A map of the united kingdom

Description automatically generated

Figure - Reference Count as a Percentage of Questions Asked per Constituency

Figure 4 suggests how the majority of constituencies do not tend to ask questions directly referencing their own constituency.

## Evaluation

The model assumes that a question containing an entity located in the constituency of the questioner implies the question is about the identified entity. This does not take into consideration the semantics. For example, a significant number of direct references included questions containing a request for response to a particular issue, whereby the constituency name is referenced.

Additionally, the model relies on the correct spelling of entities across multiple datasets. This is a concern when dealing with WikiData which relies on open-source data contribution. Moreover, it is assumed that the coordinates and constituency labels for a given place entity are correct.

The place data consists of a considerable number of entities. As seen in Figure 2, the entities cover a large range of constituencies. Given how the model was implemented, each question’s entities had to be compared against each place entity to find matches. This significantly increased computational costs as there were 25,145 questions and 764,177 which meant a total of 19,215,230,665 comparisons. This meant that the model is not scalable to larger datasets. The correct approach would have been to use a HashMap for place lookups, which would have made the runtime significantly quicker. The issue with HashMaps is that each place would have to be assigned a unique ID to distinguish between different places with the same name. Furthermore, extending functionality to fuzzy matching would require storing a separate place entity for each alternate spelling which would drastically increase memory requirements. However, this method would be preferable if the model was to be used in future.

As highlighted in the model section of the report, the Stanza library was used for NER. This dataset was trained on the Ontonotes dataset and achieved the second highest accuracy, 88.8% compared with other NER libraries[9]. However, this is not necessarily reflective of our dataset. CoNLL03’s training data for English text’s is a collection of 946 articles, with 14,987 sentences[12]. The data included 7140 locations and 6321 organisations. This is quite a small sample compared to the place data gathered in this report. Additionally, the articles were collected from August 1996 to August 1997. As names are not fixed, it is very likely that some names will have changed. Therefore, the entity recognition could have been more accurate if trained on a larger corpus with more relevant data.

Geotagging required string matches against the places dataset. The main challenge with this approach is handling conflicting spellings of identical places. The comparisons were case-insensitive, to increase matches. However, they could have benefitted from further pre-processing to remove punctuation and to homogenise special characters. Although fuzzy matching was applied to the constituency comparison, the number of matches could have been improved by applying fuzzy matching to the place names as well.

# By applying LDA topic modeling and analyzing the results, what (if any) identifiable regional differences are there in the types of questions asked?

## Data Preprocessing

Latent Dirichlet Allocation (LDA) is a model concerned with two things: *psi,* the distribution of words for each topic K and *phi*, the distribution of topics for each document *i*. Therefore, first the text was stripped of all punctuation and converted to lowercase.



Figure - Most frequent words in corpus

Additionally, any accents in the text were removed. NLTK provides a standardised set of words, known as stop words, that can be removed from text as they do not contribute to the topic of a question. For example, words like “the”, “as” and “to”.

Following the first run of the LDA model, the distribution of topics were not interpretable. As such, the words were lemmatized and the stop words were extended by removing words from the list of 100 most common that did not contribute to the topics. Moreover, words with a length less than 2 were removed. In an attempt to capture semantic relationships between words, the occurrence of bigrams were calculated using Pointwise Mutual Information (PMI) to determine co-occurrence probability of words. The words were then filter for n-gram occurrences with higher than 20 frequency. The n-grams were joined with an underscore and added to the list, ensuring they would be treated as a single word in the dictionary. Finally, extremes were removed from the dictionary, this included any words occurring in 50% of all the questions and any words that occurred less than 20 times overall.

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Processing Stage** | **Uncleaned Text Data** | **Lemmatized Text Data without Stop Words** | **Text Data without Extremes** |
| **Unique Words** | 17,685 | 11,979 | 2,524 |

Table - Unique words count

As seen in Table 9, after cleaning the text, the number of unique words was reduced by 25.28%. Although, the biggest factor was removing infrequent words, which drastically reduced the unique words. Figure 5 highlights some obvious differences in topics over the corpus, as one might expect. For example, healthcare, housing, trade and more.

Once the question text had be cleaned it had to be tokenised. Each question was converted into a list of words, which was used to create a dictionary for the corpus. The dictionary assigns a unique integer to each word in the corpus. The final step of preparation converts the tokenised words for each question into their integer values according to the dictionary.

In addition, to visualise the results of the model a custom regional shapefile was created which represented the regions of England combined with Scotland, Wales and Northern Ireland. The data for the regions of England was collected from the ONS Geoportal[5]. This was aggregated with the boundaries for countries from the same data source[6] to create a single shapefile. This was later combined with the ONS constituency boundaries[4] to assign each constituency to a region.

## Model

The process involves two stages: first a model must be trained on the entire corpus of questions; then the model can be used to determine the probability distribution of each topic for a given question.

### Topic Model Training

Initially, the model was trained on the pre-processed text using the default parameters of the genism LDA model. This produced the topic distributions as seen in Figure 6.

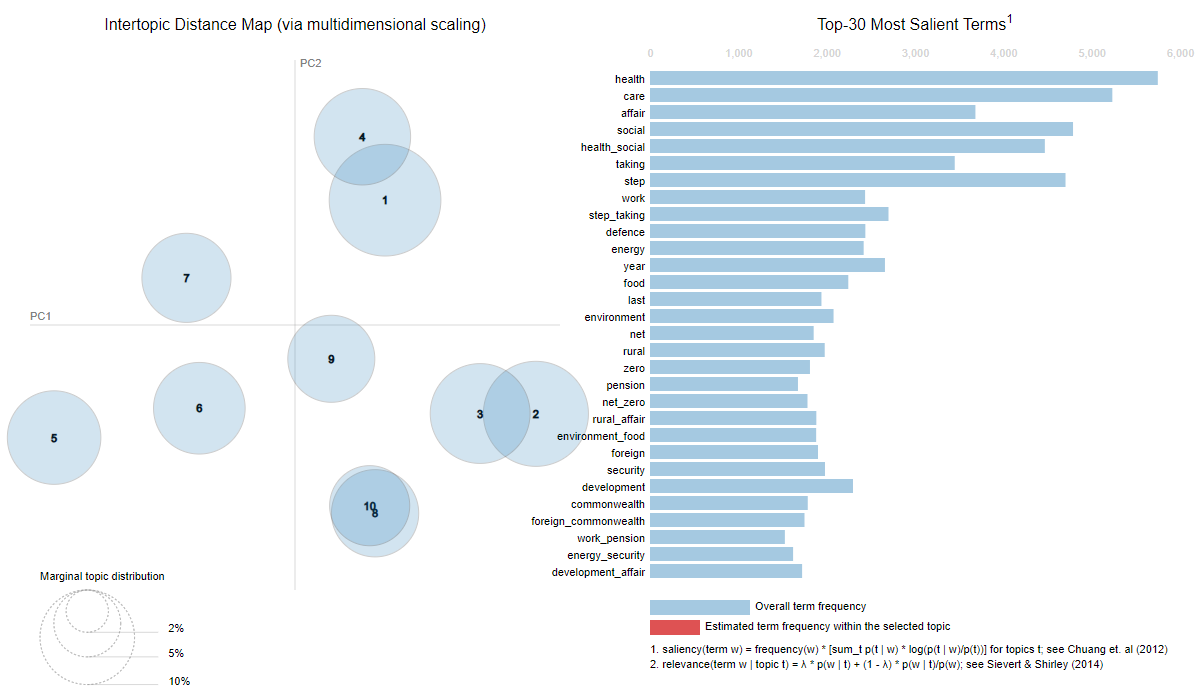
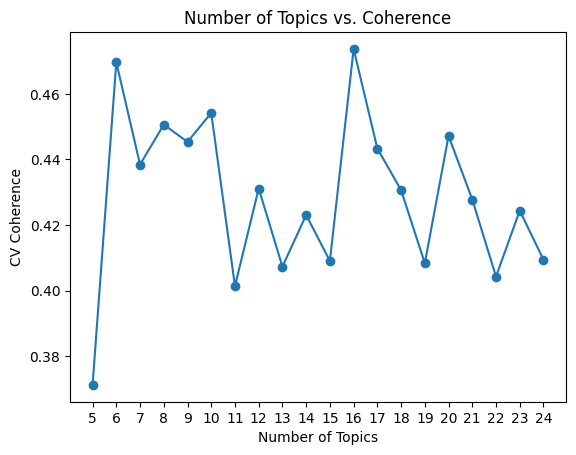


Figure – Inter-topic Distance of Base Model

The model had 10 topics to start, however there was significant overlap between the topics. Additionally, with the default parameters of the model the documents were not converging. As such the first task was determining the number of iterations the model should conduct on each document as well as the number of passes through the whole corpus. After enabling logging for the model an running it with various iterations and passes the documents would converge stably with 200 iterations and 40 passes.

Despite the model converging interpretability of the model did not significantly increase. One significant parameter for LDA’s performance is the number of topics. In order to determine this factor, the model was trained with topics between the range 5 and 24 using the CV coherence score to maximise human interpretability. Additionally, the LDA Alpha and Beta parameters were determined automatically using Gensim.

A graph with blue dots and numbers

Description automatically generated

Figure - Coherence Scores of Model

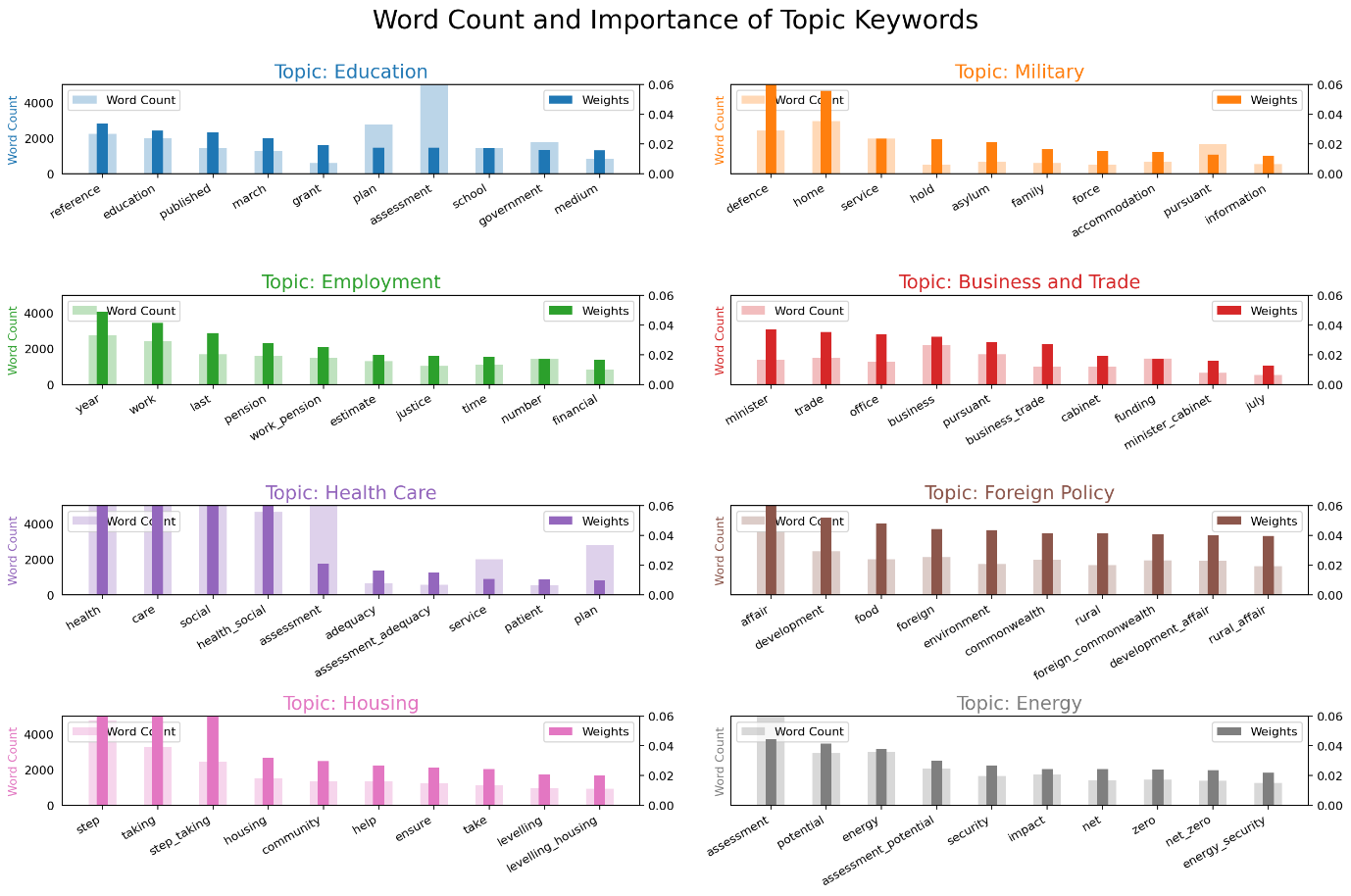
The coherence score was measured using CV and Umass, CV coherence has been shown to outperform other methods[10], however this is not always the case. As such a trade-off between both methods was used. The model with 8 topics was the most interpretable. 

Figure - Topic Breakdown of Tuned Model

As Figure 8 shows, there are clear boundaries between the words most associated with a topic. The labels have been assigned to intuitively interpret the topics.

### Assigning Question Topics

The questions obtained from the Parliamentary API included the questioner’s constituency. This information was used with the custom shapefile described in the data collection section to assign a region to each question. The topic distribution for each question and the most dominant topic were assigned.

Two separate DataFrames were created to capture the regional distribution of topics as well as the most dominant topic for each region as seen in the result section below.

### Results

This subsection discusses the results of the model. Figure 9 highlights the dominant topic by region.

A map of united kingdom with different colored states

Description automatically generated

Figure - Most Dominant Topic by Region

Five out of 12 regions predominantly asked about energy, which could be correlated with the government’s participation in the Paris Climate Accord and mounting pressure for the UK to achieve net zero emissions by 2050 (IS THIS TRUE?). However, this figure does not sufficiently capture the range of topics that MPs ask questions about.

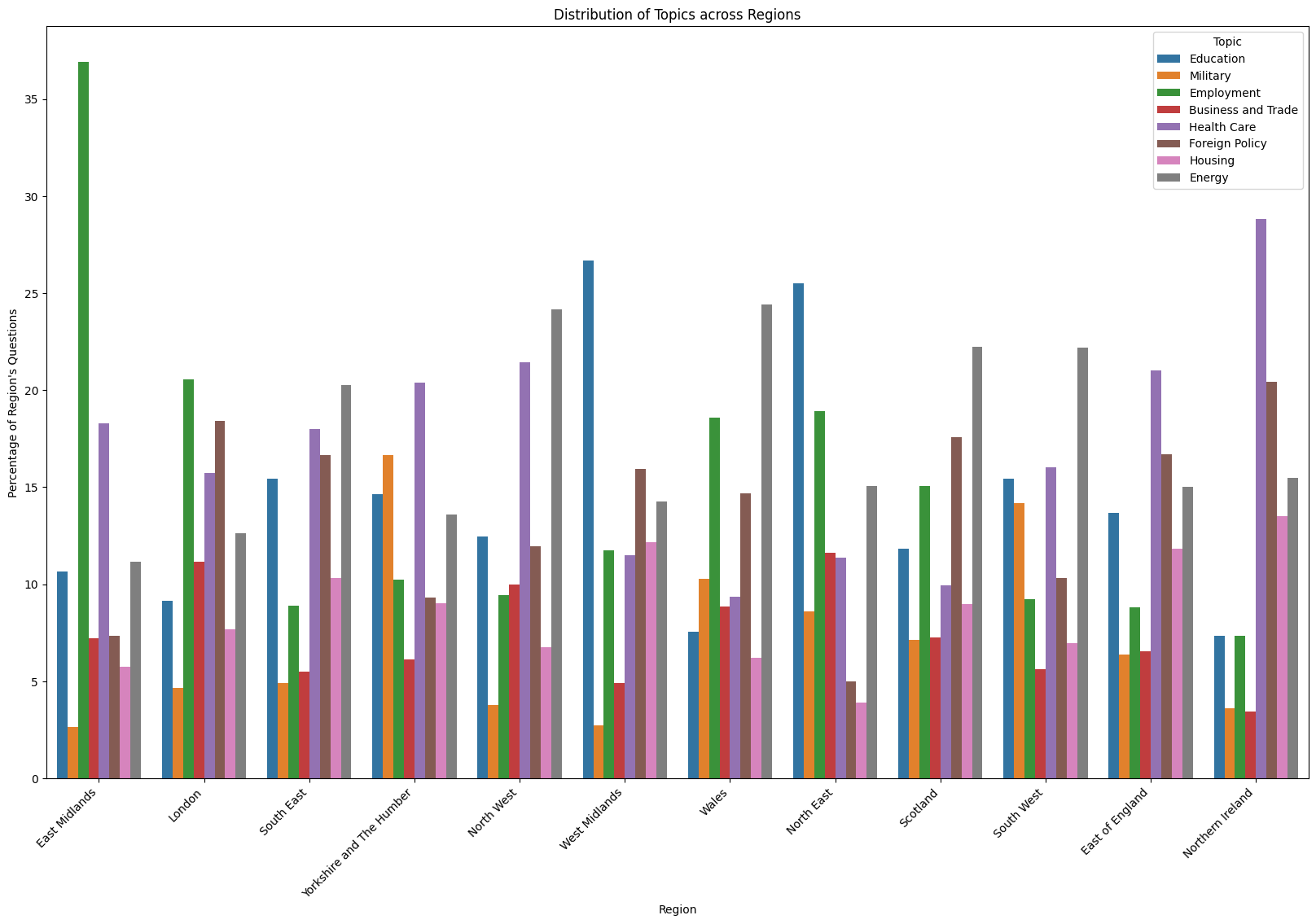


Figure - Topic Distribution by Region

As seen in Figure 10, the model suggests that the East Midlands ask the most about Employment. Northern Ireland asks a significant number of questions about Health Care. Otherwise, the most common topics of discussion are Education, Health Care and Employment.

## Evaluation

LDA models treat documents as a bag-of-words, which means that the questions are an unordered collection of words. This means that the model explicitly looks at the occurrences of specific words to determine the topic, ignoring semantics of questions. To alleviate this issue, bigrams were calculated and appended to the questions by calculating the likelihood of two words co-occurring.

The model also relies on the correct labelling of question constituency from the UK Parliamentary API[8]. Furthermore, it relies on the correct constituency names from the ONS’s constituency boundary files[4].

When tuning the model, CV coherence and inter-topic distance were the predominant measures used to determine the quality of the topic distributions. CV coherence has been shown to perform better than other established methods of coherence[10]. To further identify the quality of topics the top 30 words were considered. The issue is that it neglects many words in the given topic. Eight topics seemed the most interpretable on inspection, however there remained some topics where division into sub-topics was possible. If the corpus was larger, the number of topics could have been increased while maintaining interpretability to get a better distribution of topics.

A screenshot of a graph

Description automatically generated

Figure - UK Government Departmental Expenditure Limit 2022-23

The UK government’s[1] Departmental Expenditure Limits (DEL), highlights the main expenditures for 2023 in millions of pounds. As voters expect MPs to determine how taxpayer money is spent, government spending is a reasonable metric of what topics would be discussed in the HoC. As seen in Figure 11, the main expenditures share some correlation with the frequency of topics discussed by region. It should be noted that this data also suggests that the model is not sufficiently capturing some topics of discussion.

## Conclusion

The results of the first models suggest that MPs do not tend to ask questions referencing their own constituency. However, the data gathered did not contain an even distribution of questions for each constituency. As such, in order to generalise the results for all constituency more data is required.

The results of the second model suggest that Education, Employment, Health Care and Energy are the most frequent topics of discussion throughout the UK. This model correlates with government spending for 2023, although discussions of energy seem to be disproportionate to spending in five out of 12 regions.

## References

1. HM Treasury (2023). *Public spending statistics: July 2023*. [online] GOV.UK. Available at: https://www.gov.uk/government/statistics/public-spending-statistics-release-july-2023/public-spending-statistics-july-2023.
2. Jenny Rose Finkel, Trond Grenager and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pp. 363-370. <http://nlp.stanford.edu/~manning/papers/gibbscrf3.pdf>
3. Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. and Mcclosky, D. (2014). *The Stanford CoreNLP Natural Language Processing Toolkit*. [online] Association for Computational Linguistics, pp.55–60. Available at: https://aclanthology.org/P14-5010.pdf.
4. ONS Geoportal (2023) *UK Parliamentary Electoral Boundaries.* Available at: <https://geoportal.statistics.gov.uk/datasets/d66d6ff58fcc4461970ae003a5cab096_0/explore> (Accessed: 18 January 2024)
5. ONS Geoportal (2023) *UK Parliamentary Regional Boundaries.* Available at: <https://geoportal.statistics.gov.uk/datasets/9ab00f1e709b4423a9f70c748ee5951e_0/explore> (Accessed: 18 January 2024)
6. ONS Geoportal (2023) *UK Country Boundaries.* Available at: <https://geoportal.statistics.gov.uk/datasets/22423c95711849f4b36410d0cbc127dd_0/explore> (Accessed: 18 January 2024
7. ONS Geoportal (2023) *Index of Place Names Great Britain.* Available at: <https://geoportal.statistics.gov.uk/datasets/8f8b561f256b40c3a6df71e400bb54f0/about> (Accessed: 18 January 2024)
8. Parliamentary API (2021) *Api.parliament.uk*. Available at: https://api.parliament.uk/ (Accessed: 18 January 2024).
9. Qi, P., Zhang, Y., Zhang, Y., Bolton, J. and Manning, C.D. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. *arXiv:2003.07082 [cs]*. [online] Available at: https://arxiv.org/abs/2003.07082.
10. Röder, M., Both, A. and Hinneburg, A., 2015, February. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining* (pp. 399-408).
11. Ruchirawat, N. (2020). *6 Tips to Optimize an NLP Topic Model for Interpretability*. [online] Medium. Available at: <https://towardsdatascience.com/6-tips-to-optimize-an-nlp-topic-model-for-interpretability-20742f3047e2> . (Accessed: 18 January 2024)
12. Sang, E.F. and De Meulder, F., 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.
13. Shashank Kapadia (2019). *Topic Modeling in Python: Latent Dirichlet Allocation (LDA)*. [online] Medium. Available at: <https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0> . (Accessed: 18 January 2024)
14. Weischedel, R., Pradhan, S., Ramshaw, L., Kaufman, J., Franchini, M., El-Bachouti, M., Xue, N, et al. (2013). OntoNotes Release 5.0 . *The Linguistics Data Consortium*, p.21.
15. WikiData API (2012) *query.wikidata.org.* Available at: <https://query.wikidata.org/> (Accessed: 18 January 2024)