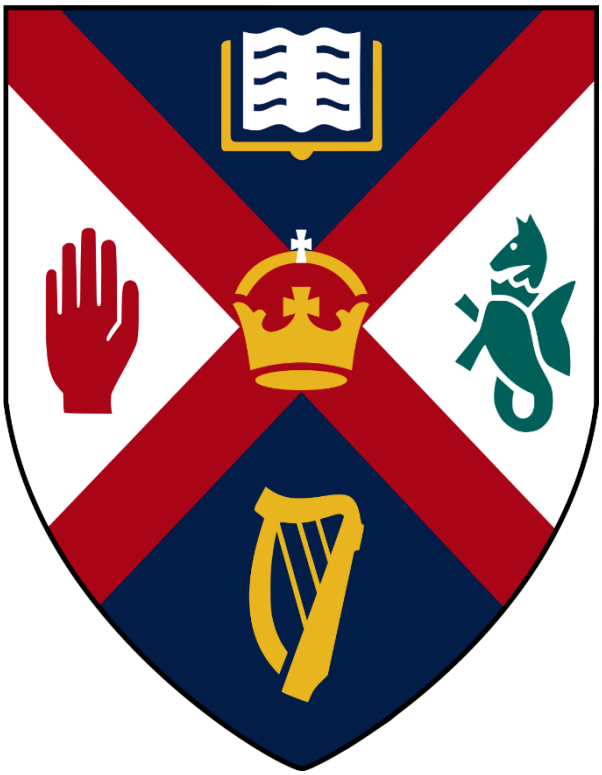

# Computational Analysis of Political Discourse

A dissertation submitted in partial fulfilment of

the requirements for the degree of

BACHELOR OF SCIENCE in Computer Science

in

The Queen's University of Belfast

by

Alexander Crowley

'02/05/2018'

## Acknowledgements

I would like to sincerely thank Dr Deepak Padmanabhan of Queen's University Belfast for supervising this project, his insight, suggestions and encouragement proved invaluable to me. Gratitude is also owed to Ishaan Jhaveri of Cornell, who kindly answered my questions about the Cornell Conversational Analysis Toolkit which section one of this project was built on top of. Acknowledgement must also be made of the staff of EEECS at Queen's University Belfast, from whom I have learned a great deal during my time at the university and who have supported me in my academic pursuits. Final recognitions must be paid to all those entities named in the bibliography of this work, without their work this project would not have such a strong background to build upon.

## Introductory Notes

This project and accompanying dissertation were focused more on research and analysis of political discourse than creating a market ready software solution. As such the software developed in the project [1] is to facilitate the analysis presented in this dissertation and following this the software specific subsections are minimised to maximise the analysis subsections, installation instructions for the software can be found in the appendices of this work. This dissertation is composed of two core sections, where the first section is the work that was originally planned for the project under the name "Templates for Answering Questions" and the second section is extra work and analysis that was performed due to the original project work being completed early.

## Abstract

Development of software for the analysis of political rhetoric was undertaken to facilitate research into the trends in political answering strength and answer consistency. The metrics generated by the software systems developed led to the conclusion that there is no statistically significant evidence that a more consistent Prime Minister is one who answers more strongly, but there was statistically significant evidence to support the notion that a Prime Minister who has answered more questions was more consistent. It was also found that there has been an upward trend in answer strength over time with only Tony Blair lying outside this trend and a general downward trend in consistency with only David Cameron improving in consistency when compared to his predecessor.

## Contents

# Section One: Templates for Answering Questions

## Problem Introduction

Questions posed by the opposition party in political settings such as Prime Minister's Questions have a plethora of elaborate and well plotted motives, and it is the primary concern of the answerer that they respond in strong, absolute terms using a template that best counters their opposition's question. Templates need not be complex, they need only place the answering party in a position of control over the topic at hand. For example, a question may be posed as "*Will the Prime Minister accept [topic at hand] will undermine the United Kingdom's security?"*, which is a prime example of a 'concede, accept' style question [2] and an archetypical answer would follow the format of *"No, I do not accept that…"* allowing the answering interlocutor to avoid conceding the point raised and disarming the asker's offensive style.

Questions need not come only from the opposition benches however, they may come from MPs residing within government benches or from parties who have aligned themselves in coalition with the government (e.g. the Liberal Democrat – Conservative coalition of 10/05/2010 to 08/05/2015) or in a confidence and supply agreement with the ruling party (e.g. the Democratic Unionist Party's agreement with the Conservatives currently in effect from 26/06/2017 to the present day). These questions have diametrically opposite motivations to those from the opposition and will often fall into the 'self-promotion' and 'agreement' categories, a trend which was noted here: *"helpful questions are often highly associated with the agreement type … reinforcing our interpretation that this type captures MPs cheerleading their own government"* [2].

Indeed, the complexity of Prime Minister's Questions has arguably increased over time as Prime Ministers must answer an increasingly wide range of question types: *"It [the data] also indicates that Prime Ministers are increasingly expected to be able to respond to a wider range of questions"* [3]. This increase in complexity necessitates a solution which can generate appropriate answer templates for a given question, allowing the answerer to avoid using an inappropriate style to respond to one question which was perfect to counter another.

Looking at Prime Minister's Questions from the Thatcher government to that of Cameron there has been a notable decrease in the number of questions being answered per session (figure 1) [3], as well as a general increase in the average number of interruptions per session and the average number of times the speaker calls the house to order per session (figure

2 [3]), *"This decrease in the number of questions correlates strongly with increases in the rowdiness of MPs ($r = -0.517$, $P < 0.01$) and the time allocated to the PM and LO ($r = -0.896$, $P < 0.01$), and less strongly, though still statistically significant, with the number of questions posed by the LO ($r = -0.342$, $P < 0.05$)"* [3] where the 'LO' is the leader of the opposition.

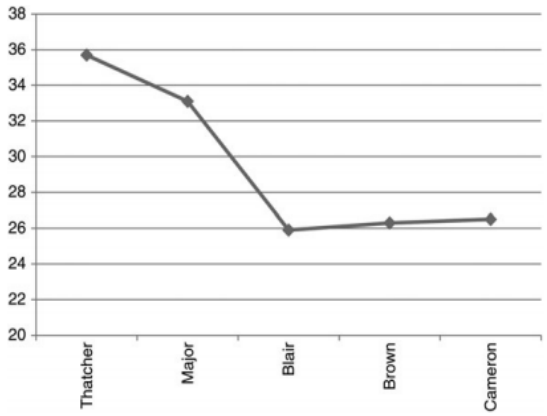


**Figure 1:** Average number of questions per session

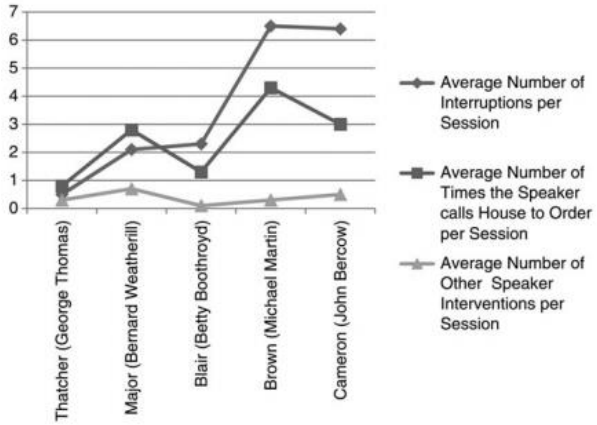


**Figure 2:** Indicators of parliamentary conduct per government with speaker in brackets

Given this decrease in the number of questions as well as the increase in parliamentary rowdiness it is now more important than ever for the answering party to appear strong and in control of each session of Prime Minister's Questions, which they can best achieve using strong answers. The requirement for strong answers is amplified further by the increased public scrutiny of Prime Minister's Questions compared to regular political proceedings, as evidenced by the fact that 'The Daily Politics Show' sees audiences of around 350,000 viewers for the Wednesday edition which includes Prime Minister's Questions compared to around 260,000 viewers for other editions [4]. To further bolster the claim that there is a high degree of public scrutiny of Prime Minister's Questions the reader is directed to this excerpt from research done by the Hansard Society: *"In our Audit of Political Engagement poll, 54% claim to have seen PMQs – either in full (16%) or in clip form (38%) – in the last 12 months"* [5].

Prime Minister's Questions is also notorious for its adversarial nature (*"I count my blessings for the fact I don't have to go into that pit that John Major stands in, nose-to-nose with the opposition, all yelling at each other"* – George H. W. Bush [6]), this only serves to back up the claim being made that strong answer templates are required to put the answering party on the forward foot. In fact the adversarial nature of discourse in Prime Minister's Questions is not merely a by-product of political animosity between the parties, *"it is both sanctioned and rewarded, a means whereby MPs may enhance their own status through aggressive face work"* [7], whereby the notion of 'face' is excellently summarised in this

snippet: *"To fail to have one's identity ratified is to lose face in an encounter, to have one's identity ratified is to have face, to maintain an identity that has been challenged is to save face. Face, then, is something that resides not within an individual, but rather within the flow of events in an encounter*" [8].

So given that the opposition party MPs will be seeking not only to undermine the answering party's face but will also be seeking to simultaneously bolster their own party's as well as their personal face it is a necessity that the answerer shut down such questions in order to not only prevent loss of face in the short term for their party, but also to avoid the rise in status of any opposition MP leading to long term issues for the answering party at the ballot boxes.

This notion of maintaining, bolstering and saving the face of the party when responding to questions from the opposition is especially important in an age where a single strong question or answer could become the next trending topic on social media, a trend parties are cognizant of and have in fact been attempting to manipulate through Prime Minister's Questions as reported by the BBC here: *"…as well as engaging with their opponent, both leaders are also trying to go viral on social media"* [9]. It is also argued that the rise in the adversarial nature of Prime Minister's Questions coincides with the televising of it: *"Arguably, this adversarial and confrontational process has only been heightened by the televising of the House of Commons (since November 1989)"* [7], so it would be expected for this trend in adversarial nature only to increase with the rising uptake of social media platforms.

On the rising uptake of social media platforms, the high usage of social media among young voters makes it a particularly important battleground for the immediate and long-term success of parties involved in Prime Minister's Questions [10], the usage of social media around the time of the Scottish Independence Referendum (dubbed 'IndieRef') was studied and the figures speak clearly to the importance of this new form of media: *"69% of members of the public (and 75% of MSYPs) reported that their use of social media to discuss political issues increased during the referendum campaign. Since the referendum, around 44% of young members of the public (and around 39% of MSYPs) report that their use of social media to discuss political issues has increased even further"* [11] where MSYP is short for 'Member of the Scottish Youth Parliament'. Hence, any Prime Minister answering a question would be well served by delivering the strongest answer in order to generate the greatest stir on social media.

Thus, with Prime Ministers facing a growing variety of increasingly adversarial questions and with their answers being scrutinised more than ever there is a clear need for a computational model that can take the format of a given question and construct a strong and appropriate answer template with which to counter the question. In the next section a solution will be proposed and its merits laid out, however it is important to keep in mind that *"all quantitative models of language are wrong - but some are useful*" [12], and with this in mind near the end of this section the potential shortcomings of the model will be discussed under the heading of 'System Evaluation & Future Possibilities'.

**Proposed Solution**

The solution proposed will take advantage of the dependent nature of answers on the questions that invoke them, this idea of dependence is summarised by Goffman in the following quote: *"Observe that although a question anticipates an answer, is designed to receive it, seems dependent on doing so, an answer seems even more dependent, making less sense alone than does the utterance that called it forth."* [13]. Previous work has shown that semantically similar questions can be paired using their answers as a component of the pairing method, further supporting the notion that similar questions lead to similar answers: *"The method can detect semantically similar questions that have little word overlap because it calculates question-question similarities by using the corresponding answers as well as the questions"* [14].

Indeed, looking back to one of the previous sources it is clear their methodology used the notion that functionally equivalent questions lead to functionally equivalent answers, where the functional nature of the questions and answers are captured by their 'fragments' and the phrasing captured by the 'motifs'. Fragments were first extracted from the questions, *"Hence, we start by extracting the key fragments within a question which encapsulate its functional nature"* [2] and then from the answers, *"In line with our focus on functional characterizations, we extract the fragments from each sentence of an answer, defined in the same way as question fragments"* [2]. Frequently co-occurring fragments were then grouped into motifs and finally the correspondence between functionally similar questions to their similar answers is noted here: *"Finally, we identify question types - broad groups of similar motifs. Intuitively, if two motifs $m_i$ and $m_j$ have vectors $q_i$ and $q_j$ which are close together, they elicit answers that are close in the latent space, so are functionally similar in this sense."* [2].

The evidence supported notion that functionally equivalent questions will lead to functionally similar answers is a core assumption of the solution being proposed, but there is

another assumption that is arguably even more important. This assumption is that, within the context of parliamentary debates, more frequently occurring answer fragments are those answer fragments which capture the functional nature of the strongest answer. This notion of frequency denoting strength is logical when considering the immediate motivations of the answering interlocutor. There is no motivation for them to appear politically weak or inept, and as such it can be assumed that those answer fragments which occur most frequently are coming from the answers that are strongest for the question at hand.

Combining these two assumptions the core concept underlying the solution being proposed is arrived at; when given a question, the strongest answer template can be derived by examining those answer fragments which most frequently occur in the answers to questions functionally similar to the provoking question, giving us a template (made up of answer fragments) which contains the functional nature of the strongest answers and which can then be filled in by the user with the relevant contextual information from the question itself such as named entities.

An appropriate data structure to store the co-occurrence frequency of any question fragment – answer fragment pair would be a matrix, allowing each unique question fragment to have a row index and each unique answer fragment a column index. The structure of such a matrix is summarised in figure 3 below, where 'cf' is the co-occurrence frequency of the given fragment pair, and 'tcf' is the total co-occurrence frequency of a given fragment.

| | $\mathbf{A_{frag\text{-}0}}$ | … | $\mathbf{A_{frag\text{-}y}}$ | $\mathbf{tcf(Q_{frag})}$ |
|---|---|---|---|---|
| $\mathbf{Q_{frag\text{-}0}}$ | $cf(Q_{frag\text{-}0}, A_{frag\text{-}0})$ | … | $cf(Q_{frag\text{-}0}, A_{frag\text{-}y})$ | $\sum_{i=0}^{y} cf(Q_{fragf\text{-}0}, A_{frag\text{-}i})$ |
| … | … | … | … | … |
| $\mathbf{Q_{frag\text{-}x}}$ | $cf(Q_{frag\text{-}x}, A_{frag\text{-}0})$ | … | $cf(Q_{frag\text{-}x}, A_{frag\text{-}y})$ | $\sum_{i=0}^{y} cf(Q_{frag\text{-}x}, A_{frag\text{-}i})$ |
| $\mathbf{tcf(A_{frag})}$ | $\sum_{i=0}^{x} cf(Q_{frag\text{-}i}, A_{frag\text{-}0})$ | … | $\sum_{i=0}^{x} cf(Q_{frag\text{-}i}, A_{frag\text{-}y})$ | |

**Figure 3:** The specification of the proposed solution matrix

This matrix would be built using a corpus of question-answer pairs from Prime Minister's Questions, the corpus used would be the same corpus used in a previously cited work [2] as

not only was it in the correct format for the Cornell Conversational Analysis Toolkit to use but it also contained ~430000 combined questions and answers resulting in ~215000 pairs for the matrix to be built on, allowing us to be confident the matrix was representative of the co-occurrences in Prime Minister's Questions. Using this matrix, the answer fragments of the strongest answer template (made up of X fragments) for an unseen question Q' could be found using a simple procedure:

1. Parse Q' and compute the set of question fragments for it (FQ').
2. For each question fragment in FQ' find the rows of the matrix for that question fragment, as seen in figure 4 where FQ' = $\{Q_{frag\text{-}x}, Q_{frag\text{-}y}, Q_{frag\text{-}z}\}$.
3. Sum the columns of each of the rows, such that the result is a single row vector (SQF') containing the summed co-occurrence frequencies, as seen in figure 5.
4. Find the X largest co-occurrence frequencies and the answer fragments relating to them (i.e. the answer fragments for the columns of the X highest values) and return these.

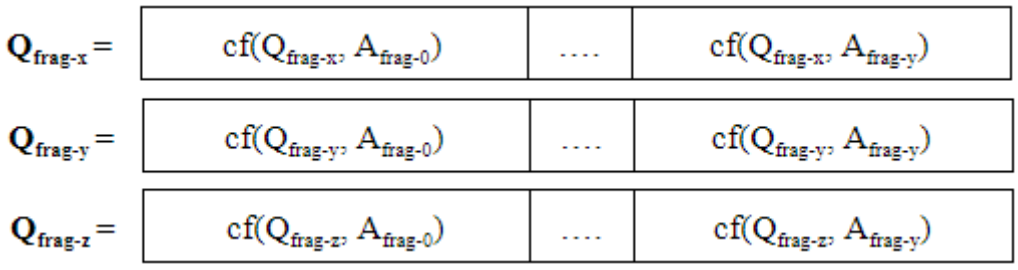


**Figure 4:** Retrieving the rows for the members of FQ'

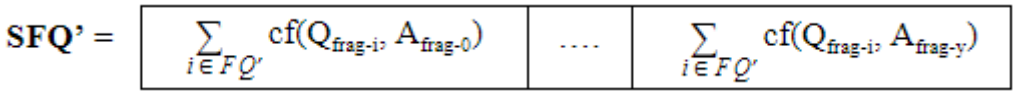


**Figure 5:** Specification of SFQ'

Of course this is an overview of the desired functionality at a high level of abstraction, there are many nuances associated with natural language processing that must be considered, an immediate example of such a consideration would be the frequency of 'stop words' (of which there exists no absolute definition, but the idea of which is well summarised in this excerpt: *"The most frequent words will most surely be the common words such as "the" or "and," which help build ideas but do not carry any significance themselves"* [15]) being very high due to their prevalence in almost all forms of language. Specifics of how stop words and other nuances are to be handled will be detailed in the 'System Requirements & Considerations' section along with details of existing code to be used or altered.

## System Requirements & Considerations

### *Matrix Dimensionality*

Given the diversity of the English language it was important to consider the dimensionality of the co-occurrence matrix that would be constructed, particularly as the number of cells in each row and column would be the largest factors impacting the efficiency of the operations being performed on the matrix. Given that the columns of the matrix would represent the unique answer fragments and the rows would represent the unique question fragments it was possible to derive the size of the matrix for different minimum frequencies, where fragments occurring less than the minimum frequency were removed. Charts examining the effects of different minimum fragment frequencies were generated as will be seen in the paragraphs that follow.

Another factor that had to be considered when experimenting with minimum fragment frequencies was that as fragments are removed there is a loss of information from the corpus. In order to visualise how much information would be lost two plots were generated which showed the distribution of answer fragment and question fragment frequencies. Figure 6 and figure 7 below show the charts that were generated.

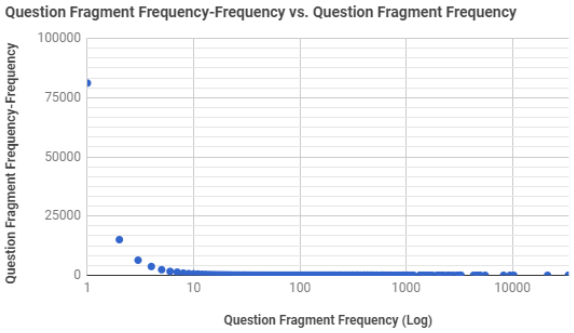


**Figure 6:** Distribution of question fragment frequencies

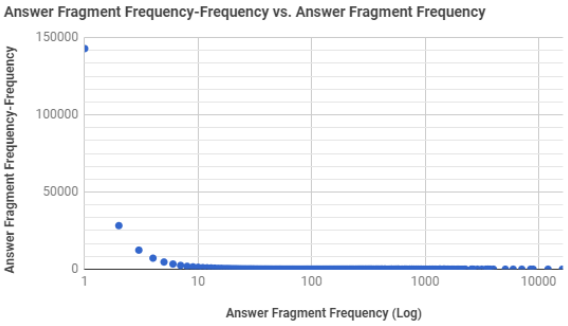


**Figure 7:** Distribution of answer fragment frequencies

As can be seen from the exponential decrease in the charts above the vast majority of the fragments for both questions and answers occur between 1 and 10 times, these low frequency fragments are not useful for analysis as they are not used in even 0.0047% (10/215000 * 100) of the questions or answers from the corpus and so add little to the potential analysis meaning the information loss is minimal. This meant that a relatively high minimum fragment frequency could be used while maintaining the vast majority of the fragments of utility to any analysis for both the questions and answers. Another visualisation of this is given in figure 8

below which shows the size of the co-occurrence matrix needed in millions of matrix cells vs the theoretical minimum fragment frequency limit used.

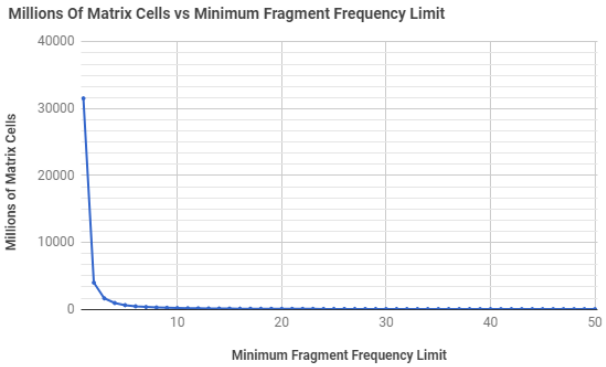


**Figure 8:** Matrix cell count vs minimum fragment frequency limit used

Again, figure 8 above shows that the vast majority of the cells are required for those fragments which occur between 1 and 10 times, this chart perfectly conveys the immense gains that could be made in reducing the dimensionality of the co-occurrence matrix. It is known that the infrequent fragments occurring between 1 and 10 times would add little substance to any analysis so using a minimum fragment frequency the efficiency of the system could be increased while maintaining the useful information. This chart also shows the impracticality of low minimum frequencies, with a minimum frequency of 1 (effectively not a minimum at all, but simply accepting all fragments as a fragment cannot occur 0 times) requiring over 30 billion cells.

### *Stop Words*

Just as fragments which occur infrequently add little to any analysis so too do those which occur an extremely high number of times. One method of counteracting extremely frequent fragments would be to have a maximum frequency limit, however with this there would be a risk of trimming out fragments that are important to a political setting such as that which the system is to be used to analyse. A superior technique, and the technique decided upon, was to remove the stop words. Previously the following definition of stop words was given as there exists no universal definition: *"The most frequent words will most surely be the common words such as "the" or "and," which help build ideas but do not carry any significance themselves"* [15].

With stop words being desirable to remove both due to their low utility in an analytical setting as well as their removal reducing the dimensionality of the co-occurrence matrix further but with no universal stop word list existing it was decided to allow the user to pass a file path

to a .txt file where each line was considered a unique stop word. This allows the user to define what they want to be treated as stop words, enhancing the flexibility of the system.

Fragments returned from the Cornell Conversational Analysis Toolkit on which the system was building had wildcards attached to them (e.g. “remind_*” or “as>*”) meaning for each stop word it was necessary to attach the two possible wildcard endings ‘_*’ and ‘>*’ so that the fragments could be easily compared to the set of stop words without having to manipulate the fragments themselves. Fragments were also often composed of two words connected by a wildcard (e.g. “how>much” or “provide_as”), in this case the fragment was divided into the two words which composed it and these were compared to the stop words set, if both were stop words then the original composite fragment was deemed to be a stop word fragment.

Thus, through this option the user could easily pass a list of stop words to be considered using only the raw words and the system would handle all the issues of parsing them and matching fragments to them.

***Single Word Fragments***

Fragments comprised of only a single word such as “remind_*” or “power>*” do not encode as much information about the structure of the text as two-word fragments such as “power_with” or “remind>fully” and thus a user may want to consider only the two-word fragments. The second obvious advantage to this is that by removing single word fragments the dimensionality of the co-occurrence matrix can be reduced further. For these reasons a boolean flag was added to the system allowing the user to trim out all single word fragments, the apt name of this flag is “remove_single_word_fragments”.

***Cornell Conversational Analysis Toolkit Modification***

The system uses the “answer_arcs.json” and “question_arcs.json” files output by the Cornell Conversational Analysis Toolkit in order to construct the co-occurrence matrix (exact details of this will be discussed in the “System Design & Implementation” subsection). These files are output by the Cornell Conversational Analysis Toolkit before the type-based clustering phase of the pipeline (details of which are in the previously cited work pertaining to the Cornell Conversational Analysis Toolkit: [2]).

Given that the system would need to deal with being given a corpus and generating the co-occurrence matrix accordingly the time intensive step of clustering within the Cornell

Conversational Analysis Toolkit would be an impediment to the system, forcing the user to wait while a step of no utility to the co-occurrence model was completed. For this reason, it was decided to modify the Cornell Conversational Analysis Toolkit by introducing a flag which made it stop after the files necessary for the matrix construction had been output, this flag is named "skip_clustering" and it is always passed as true by the system.

***Core Requirements***

This system aims to use the Cornell Conversational Analysis Toolkit to build a novel model of political rhetoric and adversarial discourse in general, although the application of it within this work is specific to political rhetoric. Thus any user of this system would be expected to understand the underlying assumptions of this model as presented in the 'Proposed Solution' section of this work, as well as having an understanding of the work related to the Cornell Conversational Analysis Toolkit [2]. With the characteristics of this typical user in mind the following requirements are defined:

- The system must be able to take the "answer_arcs.json" and "question_arcs.json" files output by the Cornell Conversational Analysis Toolkit and produce a co-occurrence matrix matching the specification previously seen in figure 3. An extension to this would be to allow the user to pass only a corpus and have the system run the Cornell Conversational Analysis Toolkit as necessary and produce a co-occurrence matrix.
- The system should be capable of outputting files that contain all the necessary information about the underlying co-occurrence matrix to reproduce said co-occurrence matrix such that an identical co-occurrence matrix could be constructed from those files.
- The system must expose to the user a parameter to allow for the passing of a stop words file, whereby each line of the .txt file will be a unique stop word that should be removed from consideration in the construction of the co-occurrence matrix. This requirement and its motivation was discussed in detail previously.
- The system must expose a parameter to allow the user to remove all fragments occurring below a defined frequency, as well as exposing a parameter to allow for the removal of fragments composed of only a single word. These requirements and their motivations were discussed previously in detail.
- There must be a flag allowing the user to specify if they wish intermediate details of the current processes to be printed to facilitate debugging, there must also be another flag

allowing the user to make the Cornell Conversational Analysis Toolkit print intermediate details of its processes.

- The user should be able to query the co-occurrence matrix for the co-occurrence frequency of any pair of fragments made up of a question fragment and an answer fragment, as well as all co-occurrence frequencies for any list of question or answer fragments with those question or answer fragments from the query not in the matrix being returned as such.
- Extending upon the previous requirement the user should be able to get a list of the most frequently co-occurring answer fragments for a list of question fragments where the number returned is defined by the user.
- Finally, the system must be able to take a user defined unseen question, parse the fragments from it and return them, allowing the user to combine this with the previous requirement to get a defined number of most frequently co-occurring answer fragments for the question fragments in an unseen question.

## System Design & Implementation

### *Constructing the Co-Occurrence Matrix*

Given the "answer_arcs.json" and "question_arcs.json" file paths a fragment co-occurrence matrix can be constructed using the following method (this method is used in the 'from_ccat_files' class method):

1. If a stop words file path was given iterate over each line in the .txt file and add the stop word with "_*" and ">*" appended to the end to the stop words set.
2. Open the answer arcs file using the given file path, and for each object in the JSON array:
    a. Remove the "span" component of the ['pair_idx'] attribute as this tells us which sentence is being dealt with which this model does not consider, it only considers the full question and answer texts.
    b. If there does not exist a mapping for the pair_idx to an array create a mapping to an empty array.
    c. Iterate over the answer fragments in the ['arcs'] attribute:
        i. If the fragment is in the stop words set, is a composite stop word fragment or is a single word fragment when the

'remove_single_word_fragments' flag has been set then skip it, otherwise append it to the array for the pair_idx.

3. Repeat step 2 for the question arcs file at the fie path provided by the user.
4. Some questions may not have answers or vice versa due to poor transcription, so taking the maps of pair_idxs to fragment arrays for the questions and answers and performing an intersection of their keys gives those pair_idxs common to questions and answers i.e. those pair_idx that have a question and an answer.
5. Remove those pair_idxs which map to an empty question fragment array or an empty answer fragment array as it is impossible to create cooccurrences in this case.
6. Iterate over all pair_idxs and iterate over the question fragment and answer fragment arrays for the pair_idxs counting the frequencies of the question and answer fragments to get total frequencies for each unique question and answer fragment.
7. Retain those fragments which occurred the minimum number of times as defined by the 'minimum_fragment_occurrence_frequency' parameter.
8. Assign ids to those question and answer fragments that were retained using maps whereby question fragment ids are in the range {0, … , number of question fragments - 1} and answer fragment ids are in the range {0, … , number of answer fragments - 1}. Question fragment ids will be their row value and answer fragment ids will be their column value.
9. Initialise an empty 2D array fitting the required dimensions of the number of question fragments retained by the number of answer fragments retained.
10. Iterate over all pair_idxs:
    a. Iterate over the question fragments for the pair_idx:
        i. If the question fragment does not have an id it has not been retained so skip it.
        ii. Iterate over the answer fragments for the pair_idx:
            1. If the answer fragment does not have an id it has not been retained so skip it.
            2. If this answer fragment was not skipped add 1 to the 2D array at location [question fragment id][answer fragment id]

This process constructs the co-occurrence matrix from the files output by the Cornell Conversational Analysis Toolkit. Some assertions and metrics have been omitted as these

ensure integrity of the system but are not requirements of the process of constructing the co-occurrence matrix.

### *Getting Fragments from Unseen Texts*

The Cornell Conversational Analysis Toolkit does not, as of the time of writing, expose any functionality to parse the fragments from text using a simple function call. As such, a workaround was devised whereby a file containing a single question and a single answer would be used to house the unseen text, parsed using a standard call to the Cornell Conversational Analysis Toolkit, the resulting files parsed in a similar fashion to that used to construct a co-occurrence matrix and the fragments returned. One advantage of this is that unseen questions and unseen answers can be parsed in a single call, allowing for double the throughput when dealing with unseen question-answer pairs. The process for parsing fragments from unseen text is as such:

1. Open the file containing the single 'dummy' question-answer pair that will house the unseen texts.
2. If an unseen question was provided replace the ['text'] attribute of the question object from the file with the text of the unseen question.
3. If an unseen answer was passed repeat step 2 for the answer object.
4. Write the newly edited question and answer objects to the original single question-answer pair file, overwriting the previous contents.
5. Pass the newly edited file to the Cornell Conversational Analysis Toolkit and allow it to run.
6. If an unseen question was provided, open the 'question_arcs.json' file and iterate over the objects, adding all the fragments in the ['arcs'] attributes into an overall question fragments array.
7. If an unseen answer was provided, do step 6 for the answer fragments using the 'answer_arcs.json' file.
8. Return a map containing the array of question fragments under the 'question_fragments' key and the array of answer fragments under the 'answer_fragments' key.

### *Getting Highest Ranked Answer Fragments for Question Fragments and Vice Versa*

In order to the get the most highly ranked answer fragments for a given list of question fragments it was possible to programmatically implement the idea of getting SQF' and finding a user defined number of maximum cooccurrence values and then finding the answer fragments

for the columns of the maximum cooccurrence values as detailed initially in the 'Proposed Solution' section. For the sake of brevity, the steps of this process will not be repeated in this section.

There is one nuanced point which appears when implementing the abstract process as detailed previously; there is no guarantee that a question fragment from an unseen question exists in the co-occurrence matrix, it could for example be one of those trimmed out by the minimum required frequency. For this reason, the function for this process returns a map specifying those question fragments not in the matrix as well as the returned answer fragments for those that were in the matrix mapped to their co-occurrence frequency in an OrderedDict retaining the order of most co-occurrence.

There is an extension to this idea that was developed as an extra feature of the system. If getting the most frequently co-occurring answer fragments for a list of question fragments is answering the question *"given these question fragments, what answer fragments should I use?"* then doing the reverse, that is getting the most frequently co-occurring question fragments for a list of answer fragments is tantamount to answering the question *"given these answer fragments, what were most likely to be the fragments of the invoking question?"*. The process of getting the most frequently co-occurring question fragments for a list of answer fragments is identical to the process defined for the opposite except that the columns of the answer fragments are summed to get SFA' (analogous to SFQ' but for answer fragments) and then the indices of the maximum values are the row indices (identical to the ids) of the question fragments.

***Plotting Results from Queries Regarding Highest Cooccurrence Frequencies***

Another extra feature that was added to the system was to allow the plotting of bar charts and pie charts to visualise the results of queries such as getting the most frequently cooccurring answer fragments for a list of question fragments. This feature simply makes use of the OrderedDict returned by such queries which contains all the pertinent information, as well as using 'matplotlib' to plot the data. An example of a bar chart plotted for the five most frequently co-occurring answer fragments for a list of question fragments (['is_aware', 'acknowledge_does']) is given below in figure 9.

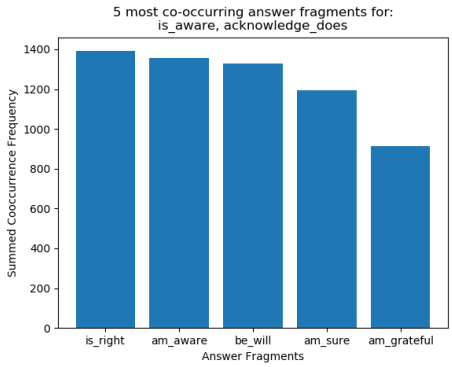


**Figure 9:** Example histogram generated for a query using [‘is_aware’, ‘acknowledge_does’]

## System Testing

Using unit testing with a defined test matrix as well as test files it was possible to cover the vast majority of the functional components of the system by first using the test matrix to validate that given a matrix in the correct format the system can perform all necessary operations correctly and then secondly using test files to test whether the matrix generated from them fits the expectations being tested. The specification for the test matrix is below in figure 10 where ‘tcf’ denotes the total co-occurrence frequency as before:

| | Answer Fragments | | | | | | |
|---|---|---|---|---|---|---|---|
| Question Fragments | / | a | b | c | d | e | tcf |
| | v | 12 | 11 | 10 | 14 | 20 | 67 |
| | w | 7 | 11 | 7 | 8 | 4 | 37 |
| | x | 3 | 1 | 0 | 0 | 10 | 14 |
| | y | 17 | 16 | 15 | 16 | 17 | 81 |
| | z | 0 | 3 | 5 | 0 | 7 | 15 |
| | tcf | 39 | 42 | 37 | 38 | 58 | |

**Figure 10:** Test matrix specification

The first set of tests on the test matrix are contained in ‘fragment_cooccurrence_matrix_tests’. After testing in detail each operation on the test matrix it is known that if a matrix can be correctly constructed then the operations using it would be performed correctly. To test building a matrix from files as well as retrieving the question and answer fragments from unseen inputs (which involves file IO hence it being coupled to these tests) the ‘fragment_cooccurrence_matrix_using_files_tests’ test file was created. Tabulated details of these tests are omitted for the sake of brevity (as mentioned in the opening notes some of the software sections have been minimised to maximise the research sections) but can be found in the ‘Testing Appendix’, it should be noted here all tests passed.

## System Derived Analysis

For this analysis the original ‘parliament.json’ corpus had to be cleaned as there were many sporadically missing tags that were required for this analysis, such as those pertaining to the party of the answering member of parliament. For this reason, a new cleaned corpus was generated by filling in the missing answer tags where necessary, for example many Thatcher and Major era answers did not include the party which was the Conservatives. Some of these answers appeared incorrectly transcribed which may have been the reason for missing tags as a way of suggesting to a user not to consider those texts, however the research script used for the analysis to follow handles those cases itself and thus a correctly tagged corpus was deemed a necessity as the analysis relied on the missing tags for party and Prime Minister matching. This cleaned corpus has the filename ‘cleaned_parliament.json’. Only answers were cleaned as only their tags were used in the analysis script.

A stop words list modified from the StandardNLP stop words file [16] was used, the modified list contains the stop words given by the StanfordNLP list but not any of the special symbols they also classify as stop words as those were not pertinent to the needs of this analysis. A minimum occurrence frequency of 200 was used, as this represents a fragment occurring in only ~0.1% of questions or answers allowing the retention of the vast majority of fragments that would have utility to the research while minimising the matrix dimensionality. Single word fragments were not removed.

Finally, with regards the parameters required for the Cornell Conversational Analysis Toolkit a ‘random_seed’ of 125 was used to allow for reproducibility and the ‘num_clusters’ used was 8, although required by the Cornell Conversational Analysis Toolkit the ‘num_clusters’ parameter is inconsequential to this research as the code skips the clustering phase using the previously discussed ‘skip_clustering’ flag.

### *Median Answer Rank Factor by Prime Minister*

In order to prevent a given Prime Minister’s own answers boosting their rank (which would lead to a definite relationship between the number of answers given and the rank determined) when analysing a given Prime Minister all questions and answers from their premiership were removed before the matrix was constructed removing any circularity in the analysis, meaning for each Prime Minister a corpus file was generated containing all question-answer pairs that did not occur during their premiership.

Another consideration that had to be made was to avoid crediting a Prime Minister for answers given by another party in the coalition as is the case with David Cameron. To this end when the answers were being analysed both the Prime Minister (given by the ['govt'] tag) and the party (given by the ['user-info']['party'] tag) were checked and if they were not a match the answer was skipped. This leads to the case where the two tags match the expected values but the answer did not come from the Prime Minister as may be the case when the Prime Minister is not present, these answers were assumed to be extremely close in style and strength to the Prime Minister's as these answers will have been drafted and laid out when the Prime Minister was aware they would not be present, so it is assumed the Prime Minister had a significant hand in creating these answers and that they are therefore representative of the Prime Minister's own answers. Given these assumptions the rank factor for a given answer was calculated as such:

1. For the question which invoked the answer to be analysed query the fragment matrix for all the answer fragments for the fragments of the question, this is retrieving SQF' as previously defined.
2. Get the co-occurrence frequency of each answer fragment from the prime minister's answer by accessing the value at the index of SQF' equal to the answer fragment id (since as previously discussed the answer fragment id is the column index).
3. Sort SQF' in descending order so that the indices of more frequently co-occurring answer fragments are lower.
4. For each answer fragment find the index of its co-occurrence frequency value in the sorted SQF' (in the case of duplicate co-occurrence frequencies the lowest index is always taken), add these indices to a 'total rank' variable.
5. Convert the total rank to be a fraction of the worst possible rank by dividing it by the number of answer fragments multiplied by the highest rank possible given by the final index of the sorted SQF'.
6. Because of the descending order of SQF' lower values of the above fraction are higher ranked (since more frequently co-occurring answer fragments are at lower indices), so to make the metrics more intuitive take the fraction away from 1 meaning highly ranked answers are now given higher rank factors and are in the range $\{0, \dots, 1\}$.

There were however many nuances that could appear while processing each question and the answer to it that could lead to a rank factor being impossible to compute. The first of these occurred when a given pair_idx did not appear in the answer_arcs or question_arcs files, this was assumed to occur when the data was incomplete as it was noted some questions and answers

appeared to be badly transcribed or missing significant sections. The next case to check was if either the questions fragments or answer fragments retrieved from the aforementioned files were empty. The second to last case occurred when none of the question fragments received from the question_arcs file were in the fragment co-occurrence matrix, this could happen when all the fragments were stop words as these had been removed from the matrix or if they had been removed from the matrix for occurring too infrequently, finally it could be the case where none of the answer fragments from the answer were in the co-occurrence matrix for the same reasons as just mentioned. In all of these cases rank factors could not be calculated and so these question-answer pairs were skipped. After all these factors were accounted for 153367 question-answer pairs were analysed across the 6 Prime Ministers, the distribution of the number of answers analysed by Prime Minister is shown below in figure 11.

| Prime Minister | Number of Answers Analysed |
|---|---|
| Thatcher | 49231 |
| Major | 25002 |
| Blair | 38721 |
| Brown | 11562 |
| Cameron | 26595 |
| May | 2256 |

**Figure 11:** Number of answers analysed per Prime Minister

One thing to note about the corpus itself is that it is comprised of 4 Conservative Prime Ministers and 2 Labour, from the corpus 153367 answers were analysed of which 103084 (67.2%) were Conservative and 50283 (32.8%) were Labour. This obviously leads to the answer fragments used by the Conservatives having a slightly higher co-occurrence frequency resulting in a Conservative bias, however an assumption is made for the purposes of this analysis that the bias is not statistically significant.

The median rank factor per Prime Minister was chosen as the best representation of a given Prime Minister's average answering strength, this decision was made because the median is more robust to skewness introduced by outliers than the mean, this robustness was important since rank factors of 1 and 0 (the best and worst possible scores respectively) occurred frequently in the case where the answer had only a single fragment which was either the best ranked answer fragment resulting in the rank factor of 1 or the worst ranked answer fragment resulting in the rank factor of 0. These extreme rank factors were not representative of any given Prime Minister's true average rank factor and as such the median was a superior choice to the mean for comparing Prime Ministers to one another.

Before discussing the trend uncovered by the analysis script described previously it is important to detail some of the key assumptions regarding the premierships and characteristics of each Prime Minister in the corpus to cement the validity and enhance the transparency of this analysis. It was assumed that all the Prime Ministers had faced an equal level of turmoil during their premierships, the tumultuous events captured in the timespan of the corpus include 'The Troubles', 'The 7/7 Bombings' and the 'Brexit' referendum among many others, with every Prime Minister in the corpus having faced one or more of these events. Another assumption that was made was that no Prime Minister was so skilled as an orator as to be completely incomparable to all other Prime Ministers, this is a reasonable assumption given that all of the Prime Ministers in the corpus have faced a great deal of criticism from a myriad of angles with there being no consensus that any one of the Prime Ministers was vastly superior to any other. The final assumption that had to be made was that the style of language used in Prime Minister's Questions had not changed so much as to render one set of years incomparable to another set of years, this is a key assumption given that if the style of speech had completely changed then it would be impossible to accurately compare Prime Ministers in two different stylistic strata as the model is very sensitive to style of speech as will be discussed later. With these core assumptions internalised the metrics derived by the analysis script were tabulated and plotted. The table is below in figure 12 and the plot is also below in figure 13.

| Prime Minister | Median Rank Factor |
|---|---|
| Thatcher | 0.8 |
| Major | 0.8001587302 |
| Blair | 0.7862068966 |
| Brown | 0.8027950311 |
| Cameron | 0.8043995244 |
| May | 0.8180102367 |

**Figure 12:** Median rank factor by Prime Minister

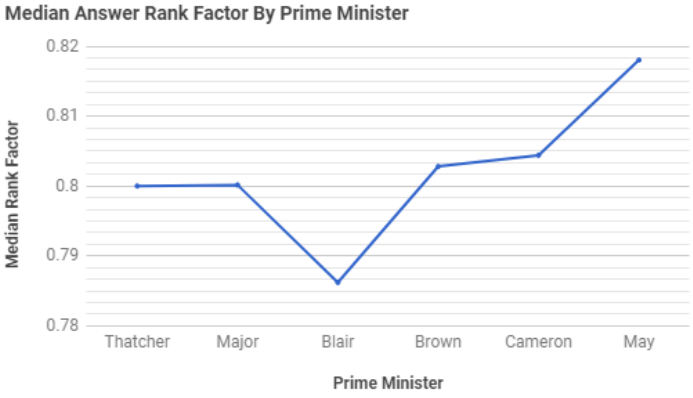


**Figure 13:** Trend for median answer rank factor per Prime Minister

In order to correctly analyse figure 13 it was imperative that outside statistical factors were controlled for where possible, obviously excluding the question-answer pairs pertaining to the particular Prime Minister ensured there was no circularity in the analysis and then secondly the number of answers analysed had to be plotted against the median rank factor to see if it was having a significant effect on the values seen. Plotting the number of answers analysed against the median rank factor for each Prime Ministers produced figure 14 below.

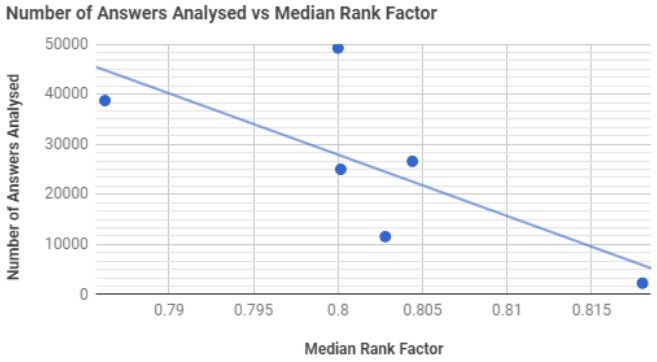


**Figure 14:** Trend for number of questions answered and median rank factor

As can be seen from figure 14 above there does appear to be a negative correlation between the two variables, however this could be tested further by calculating the Pearson product-moment correlation coefficient and investigating the statistical significance of the relationship. The r value produced was -0.7267 which confirmed that there was moderate to strong negative correlation between the number of answers analysed and the median rank factor. The next step was to calculate the p value for r = -0.7287 and n = 6, it should be noted that the null hypothesis for the Pearson correlation test is that there is no relationship between the variables i.e. r = 0 [17], and the result would be deemed significant at $p < 0.05$ leading to the rejection of the null hypothesis in this case and acceptance of the alternative hypothesis that there is a relationship between the variables. The p value calculated was 0.1023, meaning there was not statistically significant evidence to suggest there was a relationship between the number of answers analysed and the median rank factor, allowing the formation of our own hypotheses about the observed trend and results.

Before commenting on the overall trend presented in figure 13 it is pertinent to first discuss the obvious outlier: Tony Blair. Tony Blair won two successive landslide victories for the Labour party in 1997 and 2001 and won a final less decisive victory in 2005 so it may come as some surprise that a clearly talented campaigner used such weak rhetoric by the standards of the co-occurrence model. A major component of this may be that Tony Blair had a different

way of speaking than the Prime Ministers represented in the corpus used in this work, and this appears to be supported by previous work that looked at political speeches given by politicians ranging from Winston Churchill to David Cameron (the corpus has 3 Prime Ministers whose answers were incorporated into our model and is summarised in figure 15 [18] below) and compared the linguistic style of Tony Blair to these.

| Politician | Speeches | Words |
|---|---|---|
| Winston Churchill | 25 | 32,200 |
| Enoch Powell | 26 | 50,904 |
| Margaret Thatcher | 11 | 73,421 |
| Peter Hain | 6 | 16,510 |
| Margaret Harman | 6 | 15,101 |
| Hazel Blears | 6 | 17,297 |
| Gordon Brown | 8 | 31,560 |
| David Cameron | 20 | 36,674 |
| **Total** | 108 | 273,667 |

**Figure 15:** Composition of the reference corpus that Tony Blair was compared to

An example of this difference in linguistic style is Tony Blair's propensity for using the phrasing 'that is why' to introduce the perspective of the audience by answering questions they may be asking internally, Tony Blair's propensity for this style is laid out in this excerpt: *"A particular cluster that provides insight into this is 'that is why'; this occurs 36 times in the Blair research corpus but only 35 times in the larger reference corpus"* [18]. Another excellent example of this difference in linguistic style is highlighted here *"'but it was' occurs ten times in the Blair corpus but only three times in the much larger corpus for British politicians"* [18]. For further evidence the reader is directed to figure 16 [18] which compares the frequency of some of Tony Blair's style keywords with their frequency in the larger reference corpus.

| | British politicians | | Blair | |
|---|---|---|---|---|
| | Frequency | % of all words | Frequency | % of all words |
| but | 1,126 | 0.55 | 1,317 | 0.87 |
| don't | 42 | 0.02 | 144 | 0.09 |
| century | 39 | 0.02 | 112 | 0.07 |
| risk | 23 | 0.01 | 69 | 0.05 |
| deal | 44 | 0.02 | 97 | 0.06 |
| modern | 33 | 0.02 | 77 | 0.05 |
| just | 223 | 0.11 | 272 | 0.18 |
| why | 151 | 0.17 | 200 | 0.13 |
| know | 220 | 0.11 | 271 | 0.18 |
| about | 366 | 0.18 | 400 | 0.26 |
| also | 191 | 0.09 | 233 | 0.15 |
| narrative | 0 | – | 17 | 0.01 |
| today | 186 | 0.09 | 231 | 0.05 |

**Figure 16:** Keyword comparison of Tony Blair and the reference corpus

Thus, if Tony Blair did indeed have a style of speech very different to that of other Prime Ministers, as suggested by the previously cited work, and carried that speaking style into Prime

Minister's Questions it would make sense that his answers appear weaker in the co-occurrence model as the answer fragments he used may not have co-occurred frequently enough to be considered as strong as those used more commonly by other Prime Ministers. This exposes one of the core weaknesses of the co-occurrence model; the fragility of it regarding style of speech meaning a popular style will be deemed stronger than a less popular style.

Looking past Tony Blair, there is a general trend of increase in median ranking factor over time, this would suggest that Prime Ministers have become more adept at answering the questions posed to them. One hypothesis as to why there appears to be a general increase in the strength of political answers over time is the simple idea that as you iterate over the Prime Ministers in the corpus chronologically each of them has a greater number of previous Prime Ministers to look back on and learn from. Another possibility for the trend is the growth of technology during the years covered by the corpus, the core of this hypothesis is that as internet access has become more prevalent in the UK, rising from 9% of households in 1998 to 90% in 2017 [19], Prime Ministers have attempted to capture the online audience more than previous Prime Ministers, knowing that hard hitting answers will diffuse through social media at a greater rate [9] [20]. These are just hypotheses however and further research would be required to determine if they are grounded in reality.

## Model Evaluation & Possible Future Improvements

One of the major shortcomings of the model and system discussed thus far is that it relies solely on the notion that more frequently occurring answer fragments are stronger, and while in the section titled 'Proposed Solution' the evidence for this approach was presented it is still a naïve model, ignoring entirely the content of the answer beyond its functional nature. The model also ignores the non-functional content of the question, so operates purely on the functional nature of the corpus it is built upon without regard for linguistic nuances.

While the assumption of similar questions giving similar answers is supported by the literature (please refer to the 'Proposed Solution' section for discussion of this) the assumption that frequency is tied to the strength of fragments is predicated entirely on the motivation of the interlocutors being to strengthen their own position, which while a logical assumption to make in parliamentary debate the model may not adapt well to long form debates (for example Oxford style debates) or other question-answer settings such as online forums. As future possibilities are detailed in the following paragraphs the reader is invited to remember *"all quantitative models of language are wrong - but some are useful*" [12], and using this knowledge it is

possible to discuss the shortcomings and possible improvements of this model not seeking perfection, but improvement, specifically in utility towards adversarial discourse.

Without expanding the scope of the model beyond the words within answer fragments attributes of the words themselves could be considered. A previous work looking at the effectiveness of persuasion strategies on the Reddit community '/r/ChangeMyView' used several word level characteristics in their model, these were: arousal (intensity of emotion), concreteness (denoting something perceptible), dominance (expression of control) and valence (association with pleasantness). Although this work found only some of these characteristics to be useful (*"Table 3 shows that it is consistently good to use calmer language. Aligned with our findings in terms of sentiment words, persuasive arguments are slightly less happy. However, no significant differences were found for concreteness and dominance."* [21]) they could be incorporated into this model as their utility may vary depending on the corpus in use and may also be of differing utility to different users.

If for example the introduction of a concreteness factor for the answer fragments was desired, previous work [22] could be incorporated which derived concreteness ratings for 40000 English words, comprising 37058 single words and 2896 two-word phrases which could provide the rating of both single word and two-word fragments. It may also be possible to build upon previous research [23] to determine how emotive given answer fragments are by comparing them to a lexicon created to map words to the emotion associated with them (the dataset is referred to in the research as 'EmoLex'). This could be extended to also determine the emotions associated with question fragments to examine whether a given emotion in an answer best counters an emotion presented in the question, for example perhaps if anger is presented in a question it is best to respond in calmer terms to appear more in control.

Another possible factor to include at the word level would be appeals to collective pronouns such as 'we' and 'us' as research examining campaign speeches from Australian prime ministerial candidates in the 41 elections since independence found the following: *"Victors used more collective pronouns than their unsuccessful opponents in 80% of all elections. Across all elections, victors made 61% more references to 'we' and 'us' and used these once every 79 words (vs. every 136 words for losers)"* [24], hence collective appeals could be added as a factor examined per-fragment as they can be easily detected in a similar fashion to how stop words were detected through the use of a user provided list.

Looking outside the word components of any given fragment a possible improvement to the model would be for each answer fragment to have a weighting factor based on the sentences or context it is used in. Emotive language (as opposed to the emotion of single words detailed previously), for example, is a well-documented linguistic tactic used in political debates and speeches (arguably the answers given to helpful questions within parliament could be characterised as miniature political speeches in many cases) to illicit the desired response within the target audience, as evidenced by the use of emotion in political advertising: *"Our findings were fairly consistent with Brader's findings: pride was the most common emotional appeal, with 85 percent of ads containing an appeal to pride. Eighty-four percent of ads contained an appeal to enthusiasm, 48 percent contained an appeal to anger, and 24 percent employed a fear appeal."* [25], and in the language used by political candidates to increase participation among the politically engaged: *"An emotional candidate – no matter the specific emotion he expressed – increased participation amongst the most politically sophisticated"* [26]. In fact emotive language may be more important to rank than ever before in this current age of social media as research has shown that emotive language increases the diffusion of information through social networks, *"Based on two data sets of more than 165,000 tweets in total, we find that emotionally charged Twitter messages tend to be retweeted more often and more quickly compared to neutral ones"* [20], and thus an interlocutor who harnesses that potential will surely be primed to conquer the social media space.

There are many other sentence level linguistic features that could be utilised in such a rank beyond just emotive language, for example use of metaphors has also been well documented and researched as a tactic used in political rhetoric: *"the strategic use of metaphors … not only served to represent complex political issues in an easily digestible language, but also shaped and influenced the negotiations through their various mediations and the ideological intentions embedded within the metaphor"* [27] and so some answer fragments may lend themselves well to use in metaphors and this could be another factor in an improved model.

Research has been conducted into the use of hedges (defined here: *"Hedges, a specific type of qualifier, are words used to modify the meaning of a statement by commenting on the uncertainty of the information or on the uncertainty of the writers."* [28]) as a persuasive technique, it was found that hedges undermined the persuasiveness of a given statement if that statement contained facts or figures that should be unambiguous or concrete (the findings are summarised in this excerpt: *"We concur that hedges can weaken strong arguments if the hedges accompany statements of research results, presumably because research results should*

*otherwise be unambiguous."* [28]), thus, it may be possible to use this research to identify those answer fragments associated with hedges and ensure they are not returned alongside fragments associated with concreteness. Another way to incorporate this research into the model would be to look for what are referred to as 'data statements' whereby statistics are being presented to bolster a point, if a question is identified as containing a data statement then to appear as concrete as the questioner the answerer should avoid using fragments associated with hedges.

Previous work has examined many more sentence level features than those example laid out in the paragraphs preceding this to aid in predicting the outcomes of Oxford style debates, some features that were examined for the predictive model were: sentiment of individual sentences as well the sentiment transition between sentences, use of rhetorical questions and readability level [29] all of which could be incorporated as factors into the model.

There was an attempt made during the timespan of this work to create a co-occurrence matrix of 'motifs' which capture phrasings as sets of frequently co-occurring fragments of the same class (i.e. question or answer). The motifs for the questions of a given corpus are output by the Cornell Conversational Analysis Toolkit but the answer motifs are not. An attempt was made to swap the texts of the questions and their answers as a workaround to make the system output the motifs for the answers however this was unsuccessful. The code used to try this workaround is in the 'Misc' folder of the 'RhetoricAnalysis' project.

Based on the word level characteristics and sentence level linguistic features detailed in the preceding paragraphs, a question fragment - answer fragment pair could have many weighted fields based on these features that could be utilised by the user to tune the model to their needs, with the core matrix being maintained but instead of the co-occurrence frequency being the value presented in a cell (given by ($Q_{frag}$, $A_{frag}$)), it could be a rank that includes not only the co-occurrence frequency but also the factors discussed thus far, whereby the weight of each feature could be exposed to the user and made modifiable by them, as different users may have different needs based on the setting of their corpus. An example would be an emotive index that may indicate that even though a given answer fragment co-occurs frequently with a given question fragment the answers are generally negative in emotion, and thus the user may wish to avoid using the answer fragment if they are seeking a positively emotive response. This could be extended to any of the other features, for example the readability level may indicate that the sentences a given fragment appears in are often difficult for the audience to parse and should therefore be avoided despite perhaps being strong in other areas.

Another expansion in the scope of the model, beyond the sentence level characteristics, would be to investigate the topics being debated and their use. A substantial amount of previous work has been done investigating the use of topics by debaters in a variety of formats, including the 2012 Republic presidential primary debates in the US [30] as well as Oxford style debates [29]. The research in the two aforementioned settings shows that the winning side of a debate often uses topics that are stronger for them *"we find that winning sides are more likely to have used inherently strong topics (as inferred by the model) than losing sides (59.5% vs. 54.5%), a result echoed by human ratings of topics without knowing debate outcomes (44.4% vs. 30.1%)"* [29] and that those figures who appear powerful are those who introduce topics but do not shift away from the topic at hand (intuitively shifting a topic may be seen as an avoidance strategy, and research suggests the public already see politicians as avoiding questions too much *"67% of the public agree that 'there is too much party political point-scoring instead of answering the question'"* [5] so this should be avoided where possible) as evidenced here *"candidates with higher power introduce significantly more topics in the debates, but attempt to shift topics significantly less often while responding to a moderator"* [30].

These topic level features would require a change in the model away from a single matrix containing a rank made up of several ranked factors as described previously since this expansion would require the fragments to be grouped by the topics they are often used in, leading to a model made up of several matrices, one per topic, whereby each topics matrix describes the relationships between question and answer fragments within those topics, a question topic could then be derived from a given question and that topics matrix consulted for the best answer fragments. Alternatively, if a shift in topic was required (although as discussed above this is often ill advised) the matrix for the desired topic could be consulted to get those answer fragments best suited to counter the question which would hopefully negate or at least diminish the negative effect of the topic switch.

The model could be extended a step further, outside the realm of the raw text, by considering audience reactions to given questions and answers, in effect including 'audience level' features. This has already been considered in a previously cited work *("Finally, audience feedback, including applause and laughter, is also considered."* [29]) and in fact, using only audience features the outcomes of the debates could still be predicted in a majority (although a small majority) of cases with the accuracy being reported as 58.5% [29]. Audience applause and laughter levels could be easily included in the model as a weighted factor, and it may be found that these levels are correlated with those answer fragments which make appeals to the

audience or are particularly strong. Another possibility would be to examine the real time reactions of audience members based on their own political priorities, this has been previously done in research using a mobile app during the first debate of the 2012 US presidential election cycle [31]. These reactions and priorities could then be used by a given party to tailor their answers towards using fragments which play well with their core supporters or could perhaps help a party use fragments that will engage undecided voters.

An obvious issue with the model is that it does not consider the fact that a question may be linked to previous answers and indeed also to previous questions, the co-occurrence matrix model has no concept of the chronology of the question-answer pairs and does not try to link them beyond the limited scope of their functional components (and even when expanded as detailed in the paragraphs preceding this one it would still not chronologically link a series of questions and answers). It could, for example, be the case that a speaker delivers a series of deliberately leading answers to corner the questioner, yet the co-occurrence model would not be able to uncover this tactical use of concessions. Indeed the use of concessions is not only applicable to debates but to negotiations of all sorts (and arguably debates could be framed as negotiations, where both sides wish the other to concede to their point of view) and has been characterised as an extremely effective tactic: *"The truly gifted negotiator, then, is one whose initial position is exaggerated enough to allow for a series of concessions that will yield a desirable final offer from the opponent, yet is not so outlandish as to be seen as illegitimate from the start"* [32].

This could perhaps be achieved by using topic tracking (which is a heavily researched field within natural language processing [33] [34] [35] and was used in works previously cited in discussing the shifting of topics as a feature to be examined [29] [30]) combined with tracking the rank factors of the answers to a series of questions on the same topic. Tracking these two features chronologically may reveal a series of low to medium strength answers on a given topic being followed by a resoundingly strong answer. This tracking could reveal which answer fragments are best for presenting leading answers, and which are best for a strong finishing answer. This goes beyond the simpler extensions of the model discussed previously and would require a tagging system whereby fragments could be tagged as having features such as 'weak_leading' to indicate a given answer fragment is weak but is often used in a leading style. This tagging system could replace parts of the weighted factors implementation where labels are more appropriate than weights (for example the strength of particular emotions as modelled by a weighted factor is going to be subjective and therefore dependent on the user, whereas

atomic labels such as ‘sad’ are more easily understood by humans) as it would allow the user to search for atomic labels instead of having to finely tune their parameters. This tagging would also allow migration away from matrices completely, allowing the co-occurrence frequencies to simply to be an array or map tagged onto each fragment permitting easier storage in forms such as JSON as all fragment features could be encoded as tags.

There are a myriad of forms this system could take when evolved to such a level of complexity as has been arrived at in this discussion, and even beyond what has been discussed there are a plethora of possibilities, for example those fragments most quoted in positive tweets could be tracked and this given a weighting, but there is a point at which the growth of the model exceeds what is achievable within a reasonable time frame and instead jumps into the realm of idealism and not realism. This is perhaps the a reason why *“all quantitative models of language are wrong - but some are useful*” [12] and given this the most achievable form this model could take in the near future would be the multiple matrices model (where there is a matrix per topic discussed) combining multiple weighted word-level factors as discussed previously.

**Section One Conclusion**

The system designed and implemented in section one of this work correctly models a co-occurrence matrix to the specifications laid out initially in figure 3. This system fulfils the functional requirements of being able to derive the most frequently co-occurring answer fragments for lists of question fragments and vice versa, as well as being able to compute the fragments in previously unseen questions and answers to suggest what fragments the best answer will contain or what fragments the provoking question was likely to be composed of.

The co-occurrence matrix model underlying the system was used to investigate the trend in answer strength over time by using the novel rank factor metric. It was determined that Prime Ministers were generally becoming more adept at answering questions over time, apart from Tony Blair whose rhetorical style may have exposed the underlying weakness of the co-occurrence model in that popular styles are assumed to be stronger. There was no statistically significant evidence found to suggest the rank factor was affected by the number of answers analysed for each Prime Minister and thus that was ruled out as the reason behind the observed trend. It was postulated that the trend could simply be because as each successive Prime Minister enters their premiership they have a greater number of Prime Ministers to look back on and draw insights from. Another possible reason was that in the age of social media it has

become obvious that hard hitting answers diffuse best and thus Prime Ministers have been seeking to give answers that will diffuse rapidly on social media [9] [20] over time as internet access and thus social media usage have increased across the UK with time [19].

Many possible improvements to the co-occurrence matrix model were discussed, with a system allowing there to be multiple co-occurrence matrices each representing a topic with the values within them being weighted using word-level factors such as concreteness, being concluded as the most realistic improvement to the system within a reasonable time frame.

## Section Two: Analysis of Alignment of Political Questions with their Answers

### Research Introduction & Proposed Solution

In section one of this work the fragment co-occurrence matrix made use of the assumption that similar questions would lead to similar answers, this assumption was supported by evidence that was cited in the relevant subsection [13] [14]. However, it was decided to empirically research whether this assumption held in the political setting of section one and investigate whether a quantitative metric could be used to derive a relationship between the observed median rank factors for the Prime Ministers and their 'consistency' as measured by this quantitative metric of similar questions leading to similar answers.

Previous research has suggested that political leaders who change their positions on issues they have previously presented as moral issues suffer a detrimental effect in the public perception of them: *"Leaders who changed their moral minds were seen as more hypocritical, and not as any more courageous or flexible, than those whose initial view was amoral. They were also seen as less effective and less worthy of support, and indirect effects suggested that these effects were due to the effects on hypocrisy"* [36]. This lends credence to the idea that an inconsistent politician may be a weaker one, particularly with morality and references to religion being growing forces in the sphere of UK political rhetoric: *"Despite 9/11, there is a general upward trend in religious rhetoric. Prior to 2001 (i.e. 1998-2000) there are, on average, 11 references and allusions made in party conference speeches per year, whereas after 2001 there are over 16.5"* [37]. Indeed, given that the median rank factor for each Prime Minister was previously calculated if the 'consistency' of each Prime Minister was also calculated it would be possible to investigate whether the two are correlated in a statistically significant way using a Pearson product moment correlation test.

Previous work has also indicated that self-identifying liberals are more consistent in their beliefs than self-identifying conservatives*: "Across diverse samples and attitude measures, liberals were typically higher in ideological consistency than conservatives"* [38]. This could lead to an examination on whether Prime Ministers from the traditionally more liberal Labour party were more consistent than those Prime Ministers who were members of the traditionally more conservative, aptly named, Conservative party. This would assume that the consistency in viewpoint presented in the aforementioned work would be reflected in a consistency in the answers given to similar questions, which is likely to hold as the political ideology of the party will be reflected in their stances on many of the issues they are being questioned on.

Thus, calculating an empirical metric of consistency in political answering will not only allow reflection on the assumption made with the fragment co-occurrence matrix seen in section one whereby the assumption was that similar questions would lead to similar answers, but it will also allow comparisons to be made to the results of the previously cited works of this sub-section.

In order to empirically evaluate whether similar questions lead to similar answers it is imperative to be able to represent the question-answer pairs in a multidimensional space, requiring the derivation of feature vectors for the question and answer components of the pair. Many techniques exist for converting text into feature vectors, the most common of which is the bag-of-words model as it is efficient and easy to interpret as the core concepts underlying its implementation are simple, however this comes with the disadvantage of the order of words being lost resulting in very different sentences having identical representations [39]. A representation of the questions and answers that can make use of the semantic relationships between words as well as dealing with the relatively high length of political questions and answers is required. This requirement is fulfilled by a technique referred to as 'Paragraph Vector', the paper in which this method is presented makes clear that it is applicable to any length of text: *"Unlike some of the previous approaches, it is general and applicable to texts of any length: sentences, paragraphs, and documents"* [39].

'Paragraph Vector' or as it is more commonly known 'Doc2Vec' builds upon previous work in representing words as vectors in order to predict the next words in a sentence, known as 'Word2Vec' which was presented in the following work: [40]. The link between Doc2Vec and Word2Vec is summarised here: *"Our approach for learning paragraph vectors is inspired*

*by the methods for learning the word vectors. The inspiration is that the word vectors are asked to contribute to a prediction task about the next word in the sentence. So despite the fact that the word vectors are initialized randomly, they can eventually capture semantics as an indirect result of the prediction task. We will use this idea in our paragraph vectors in a similar manner. The paragraph vectors are also asked to contribute to the prediction task of the next word given many contexts sampled from the paragraph"* [39].

One of the core advantages of Doc2Vec which shall be of maximum utility to the solution to be proposed in this section is that semantically similar words are closer in the feature space [39] [40], for example "frail" and "weak" will be closer in the feature space than "frail" and "strong". This semantic distance in the feature space allows us to make the inference that although questions and answers may not be posed using exactly the same language or words the semantic similarities will lead to them being closer in the feature space.

Distances within a feature space have been previously used to determine whether similar problems have similar solutions within a case reporting system used in the European Space Agency [41] and a similar reporting system used within the health and safety domain [42]. This notion of determining whether similar problems get similar solutions was dubbed as the 'alignment' of a problem and its solution. The alignment of a case is determined by the propensity for a problem-solution pairs nearest neighbours in the problem space to also be its nearest neighbours in the solution space (as visualised in figure 17 [41] below).

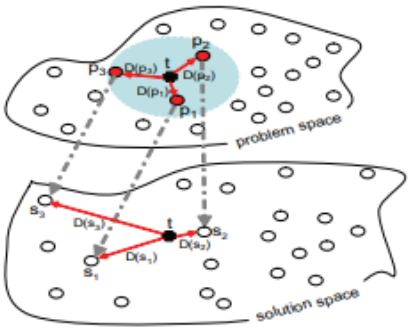


**Figure 17:** Visualisation of nearest neighbours in the vector spaces

The solution proposed will build upon this idea, creating a feature space for questions and their answers through use of Doc2Vec and then calculating the alignment using the equations presented in the previously cited paper (given later in this subsection) dealing with alignment whereby the 'problem space' in figure 17 will be the 'question space' in the model underlying this system and the 'solution space' will be the 'answer space'. The feature space will be

generated using the same parliamentary corpus as used in section one, giving an extremely large breadth of question-answer pairs as well as a diversity in the parties and rhetorical styles being represented and allowing a fair evaluation on whether alignment affects answer rank factors.

Once a feature vector space is generated it will be simple to calculate the alignment of any question-answer pair, affording the opportunity to investigate whether particular Prime Ministers were more consistent in their rhetoric by analysing the pairs in the corpus pertaining to their parliamentary term(s), whether this had any effect on the median rank factor observed for the given Prime Minister in section one as well as allowing an empirical evaluation of the previous assumption that similar questions will lead to similar answers. As with the co-occurrence matrix analysis there will be a need to check both the government and the party of the answers to ensure credit is not given to a Prime Minister for answers given by a coalition party.

One of the previously cited works pertaining to the alignment formulas has some ambiguity in its description of the calculations, the main ambiguity is introduced here: *"A distance function D(t, $p_i$) or D(t, $s_i$) measures the distance between t and $c_i$ in either the problem or solution space giving a value between 0 and 1"* [41]. Given that the distance functions must return values between 0 and 1 they must return normalised distances, meaning there needs to be a maximum and minimum distance to use in the normalisation. However, it is not obvious whether these should be taken from within the k-nearest neighbours or whether they should be taken from across the entire space for the given category i.e. should the maximum distance of 1 between 't' and a given problem be the distance between 't' and its furthest nearest neighbour or the furthest problem within the entire vector space?

$$Align(t, c_1) = 1 - \frac{(D(t, s_1) - Ds_{min})}{(Ds_{max} - Ds_{min})}$$

$$\text{CaseAlign(t)} = \frac{\sum_{i=1} (1 - D(t, p_i)) * Align(t, c_i)}{\sum_{i=1} (1 - D(t, p_i))}$$

**Figure 18:** Equations to calculate alignment of a problem-solution pair

Given the equations (taken from the same work as the ambiguity originates in [41]) as presented above in figure 18 if it is the case that the maximum distance of 1 should be the distance to the furthest nearest neighbour then the 1-D(t, $p_i$) factor in the calculation of CaseAlign(t) will be 0 for the furthest nearest neighbour meaning the entire top line is 0, meaning the alignment of the furthest nearest neighbour is not considered in the final formula. This is not a major issue, but

it does present a peculiarity whereby making the calculation of alignment using k-nearest neighbours only actually uses k-1. There is an advantage to this method over considering the entire space which is that normalising the distances within only the k-nearest neighbours means the distances are much faster to calculate for each point as opposed to having to renormalize over an entire vector space for every point considered. Despite the advantage of normalising within the nearest neighbours the system presented in this section uses the entire space of the corpus given to normalise within as there is no ambiguity in this case about the value of k being used not matching the number of neighbours actually used in the alignment calculation.

Given that the abstract ideas underpinning the system to be developed have been discussed as well as the mathematical ambiguity of the underlying equations resolved the reader is directed to the sections following this whereby the more nuanced programmatic details of the system will be explored and discussed.

## System Requirements & Considerations

### *Stop Words*

As in section one, stop words will add little to any analysis and thus can be passed to the system using a .txt file where each line is a stop word. Upon all the text documents from the corpus having been loaded and stored they are iterated over and the stop words removed, the order of the non-stop words is preserved as word order is important for the Doc2Vec model underlying the system to build its predictive model. Without order preservation, as previously discussed, two sets of text using similar words in completely different orders could end up by chance having the same random ordering (this could occur for example if a set removal was used to remove the stop words as the resulting set of non-stop words is not ordered) resulting in them being close in the vector space artificially.

### *Core Requirements*

This system is more restricted in its requirements than the system developed in section one as it is not designed to create a novel model of political rhetoric and adversarial discourse but instead to facilitate research into a specific notion: rhetorical consistency. A typical user of this system would be expected to have a specific research goal in mind which necessitates the use of alignment calculations as a measure of consistency, this also implies some knowledge of natural language processing and an understanding of the underlying Doc2Vec model built

using the Gensim [43] Python package. As such the core requirements of this system are few, and are detailed below:

- The system must allow the user to generate a feature vector space for a corpus of questions and answers where the corpus is a JSON array of question and answer objects. All objects must have [‘id’] and [‘text’] attributes and the answers additionally must have [‘reply-to’] and [‘is_answer’] attributes whereby the [‘reply-to’] attribute is the [‘id’] attribute of the invoking question.
- The system must also allow the user to specify a .txt file path containing stop words they wish to be removed from the question and answer texts. The Gensim Doc2Vec core parameters of ‘vector_size’, ‘min_count’, ‘epochs’ and ‘workers’ should also be exposed to the user to be tuned as appropriate to their needs.
- The system should expose a ‘verbose’ flag allowing the user to specify whether the system should print intermediate details of the processes being performed. Such information could be used for debugging or diagnosing data issues.
- The system should allow the user to calculate the alignment of any question-answer pair by passing the question id and answer id. The value of k to be used in the calculation of the alignment should also be exposed as a parameter to the user.
- Building upon the previous requirement the system should allow the user to calculate the alignments of all question-answer pairs in their corpus using a single function call. This function must also expose the value of k to be used as a parameter. The alignments calculated should be stored in a list which is in the same order as the question and answer lists meaning the alignment at index ‘i’ is the alignment of the question answer pair made up of the question at ‘i’ and the answer at ‘i’.

## System Design & Implementation

### *Constructing the Feature Vector Space*

Given a corpus file path the system developed in this section generates a feature vector space containing a single feature vector for every question and answer, the corpus provided should be a JSON array of question and answer objects. The process for generating the feature vector space is provided below and the JSON tags used are mentioned where appropriate:

- If a stop words file path was given initialise the stop words set using the data at the given file path.

- Load the corpus object as a JSON array from the provided file path
- Iterate over the objects in the loaded JSON array:
    - If the objects [‘is_answer’] attribute is true, then create a mapping in the question to answer map from the objects [‘reply-to’] attribute to the objects [‘id’] attribute.
    - Append the objects [‘id’] attribute to an array of document labels.
    - Append the objects [‘text’] attribute to an array of all documents.
- If a previously saved model file path was given load that model, otherwise clean the stop words from each text, then feed the ids and texts to a newly created model and train the model.
- Iterate over the map of question ids to answer ids:
    - Load the feature vectors for the question id and answer id.
    - Ensure these feature vectors are not duplicates of previously seen vectors by comparing them to a list of all feature vectors seen thus far, if they are raise a ValueError otherwise add them to the all-encompassing list of feature vectors previously compared against.
    - Add the question id and feature vector to question specific lists.
    - Add the answer id and feature vector to answer specific lists.

Once this process has concluded the system has an entire feature vector space for the corpus stored in easy to access data structures, this allows for an extremely easy process for finding a questions nearest neighbours by calculating the distance to every feature vector in the list of question vectors, finding the indices of the k smallest distances and then querying the question id list at those indices to get the ids of the nearest neighbours.

***Use of Concurrency for Alignment Calculations***

Given that vectors, question ids and answer ids do not change due to the calculation of alignment making the alignment calculations concurrent was a natural step as no critical sections of code would have to be specified if none of the core data structures were concurrently modified. Thus, the method ‘process_question_answer_pair_alignment’ could be used concurrently on different pairs because it only reads from shared resources and only ever operates on distances which are not shared between threads meaning even in the case that entities had overlapping nearest neighbours there was no possible conflict that could result in adverse effects such as race conditions or lost updates.

Given that the nearest neighbour search extends across the entire space for both the questions and answers it follows that the complexity of the operation is dominated by O(num_answers * num_answers + num_questions * num_questions) which given that the number of questions and answers are equal is O(2 * num_questions * num_questions), replace 'num_questions' with 'n' and dropping the constant 2 gives a complexity of $O(n^2)$.

Obviously splitting the computation of the alignment of a list of question-answer pairs over multiple threads does not reduce this computational complexity, however it does certainly speed up computation compared to a single thread operating. The analysis time compared to the number of question-answer pairs analysed is below in figure 19, these times were derived during the analysis of median alignment by Prime Minister which will be discussed later and were generated using 8 threads on a system the specifications for which can be found in the appendix.

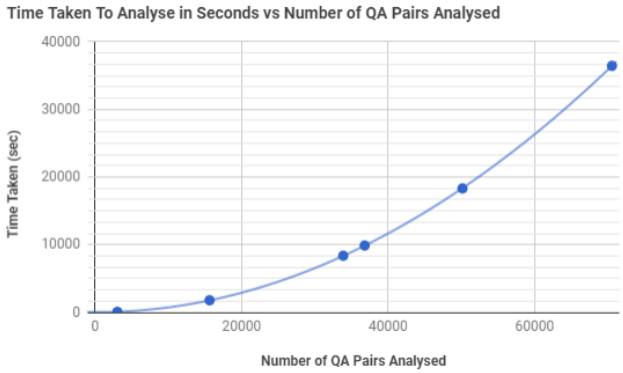


**Figure 19:** Time taken to analyse all question-answer pair alignments versus the number of pairs analysed

Adding a polynomial line of best fit of second order perfectly shows this $O(n^2)$ complexity. As can be seen computation times were still extremely high stretching into several hours as the number of question-answer pairs analysed grew despite the use of concurrency. The concurrency is achieved using the 'starmap' function from the Python multiprocessing library. A convenient feature of this function is that despite the fact the processes are performed concurrently the order of outputs in relation to the inputs is maintained [44], so if the corpus was passed in chronological order the alignments calculated would also be in chronological order, possibly allowing for fine grain timescale analytics.

## System Testing

Unit testing was used in conjunction with SimpleNameSpace [45] to mock the Gensim Doc2Vec model, as well as with the test vector space generated by using the 'testing' flag of the QAPairAlignmentCalculator to validate the system. The test cases from 'qa_alignment_calculator_tests' are described in the appendix since as mentioned before some sections have been minimised in the main text to allow for more in-depth research and theory sections, it should again be noted that all tests passed.

## System Derived Analysis

Before discussing the research and results generated it is pertinent to note that the parameters used for the Gensim model which had been trained on the entire corpus were those given as defaults by the system, these are a 'vector_size' of 100, a 'min_count' of 200 (matching the minimum frequency used in the research of section one), 100 'epochs' and 8 'workers'. It should also be noted the same cleaned corpus generated for the research of section one was used here as the basis of the corpuses generated for each Prime Minister. The same stop words file used for the research of section one was used in this research also to maximise the comparability of the results. A 'k' value of 5 was passed for the alignment processing and 8 threads were used on the same system as previously mentioned.

### *Median Alignment by Prime Minister*

Each prime minister was examined in turn, and a corpus was generated for them (which was a subset of the previously mentioned cleaned corpus used) that contained only the questions and answers from pairs where the government was that of the prime minister in question as well as the party, as previously mentioned both of these checks are required to prevent, for example, David Cameron's statistics being skewed by including those of Nick Clegg due to their coalition, this corpus was then used with the 'qa_alignment_calculator'.

It is important to note that because the corpus being passed contains only the question-answer pairs pertaining to the Prime Minister in question it means the distance normalisation across the entire corpus space is specific to the given Prime Minister, so the distances are only normalised within the space of their own corpus, resulting in a better intuition of the 'consistency' of a given Prime Minister as it would be unfair to use the pairs of another Prime Minister during normalisation as a Prime Minister should be considered consistent relative to their own rhetoric and not the rhetoric of another Prime Minister.

Before entering discussion about the median alignment values calculated for each Prime Minister, or how they relate to the previously calculated median rank factors, it is important for us to examine whether the number of question-answer pairs analysed had a statistically significant effect on the alignment values observed. Figure 20 below shows the relationship between the number of question-answer pairs analysed for each Prime Minister and the median alignment calculated.

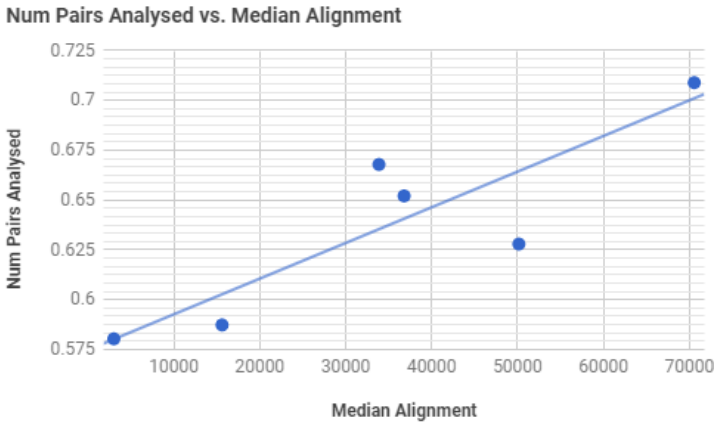


**Figure 20:** Chart of the number of pairs analysed versus the median alignment calculated

There does appear to be a strong positive relationship and a Pearson product-moment test gave a correlation coefficient (r) of 0.8756249926 which subsequently gave a p-value of 0.02225 at n = 6. Thus, there is a statistically significant correlation between the variables at the 0.05 level, leading to a rejection of the null hypothesis that there is no correlation.

Of course, it cannot be stated for certain say there is a causal relationship between these variables, however there may be an explanation for the strong positive correlation observed. If the assumption is made that there are a finite number of topics a Prime Minister may be questioned on it follows that those Prime Ministers who answer more questions are going to generally have more answers within each possible topic. This would therefore lead to the nearest neighbours in the question space being much more likely to be on the same topic as the current question under examination, which intuitively would lead to a higher alignment as the answers to those questions will therefore be closer in topic and semantics to the answer to the original question by virtue of them being on the same topic.

The inverse of this of course is that a Prime Minister who has not answered many questions will be more likely to have questions not on the same topic within the nearest neighbours of the current question under examination, leading to answers that are distant from the answer to the examined question resulting in a lower alignment. A possible experiment to

determine whether this is the case would be to perform this same analysis with varying values of k to examine whether those Prime Ministers who have answered more questions see a lesser drop in their alignment as the k value increases as it takes more neighbours for them to start counting pairs that are on different topics. It may of course simply also be the case that Prime Ministers who answer more questions learn to be more consistent over time.

Although there does appear to be a relationship between the median alignment and the number of pairs analysed it was still investigated whether a higher alignment lead to a higher rank factor since it was previously shown there was no statistically significant relationship between the median rank factor and the number of answers analysed. Thus, any relationship observed between alignment and rank factor would not be due to the number of pieces of text analysed for both. The table of figure 21 below shows the median alignment values for each Prime Minister and the chart of figure 22 below shows the median alignment and median rank factors per Prime Minister plotted.

| Prime Minister | Median Alignment |
|---|---|
| Thatcher | 0.7089139178 |
| Major | 0.667845586 |
| Blair | 0.627882604 |
| Brown | 0.5872814814 |
| Cameron | 0.6520517195 |
| May | 0.5803157998 |

**Figure 21:** Tableted median alignment by Prime Minister

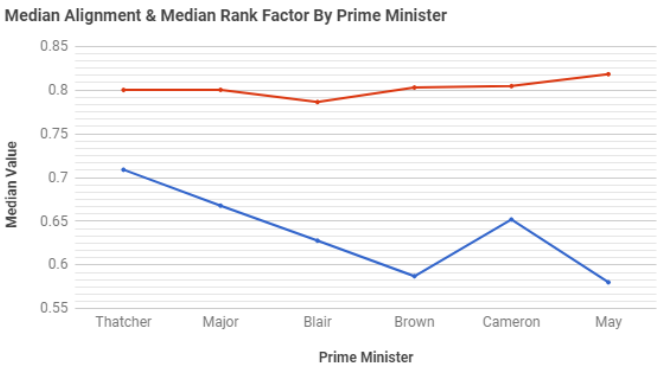


**Figure 22:** Median alignment charted with the median rank factor

The first notable inference to be drawn from the metrics is that all the median alignments are greater than 0.5, meaning generally a Prime Minister is more likely to give a similar answer to

a similar question meaning the assumption of this being the case as made in section one generally holds.

As can be seen from figure 22 the trend in median alignment appears completely disparate to the trend in median rank factor. Our hypothesis on the relationship between rank factor and alignment was that those Prime Ministers who were more consistent as measured by the median alignment would have generally higher rank factors. This would be caused by the fact that a higher median alignment meant the Prime Minister in question was using answers semantically closer to others in the corpus (and therefore closer in the feature space, hence higher alignment values) that were generated by similar questions, and it is known that a major disadvantage of the co-occurrence matrix is that it favours those fragments most commonly used, which would be the most frequently used semantic elements, so those Prime Ministers using the most common semantic elements to answer a question would rate highly in both alignment and rank factor. Figure 22 above appears to show little to support for this trend however it was important to test this empirically, plotting the median alignment and median rank factor against one another results in the trend seen in figure 23 below.

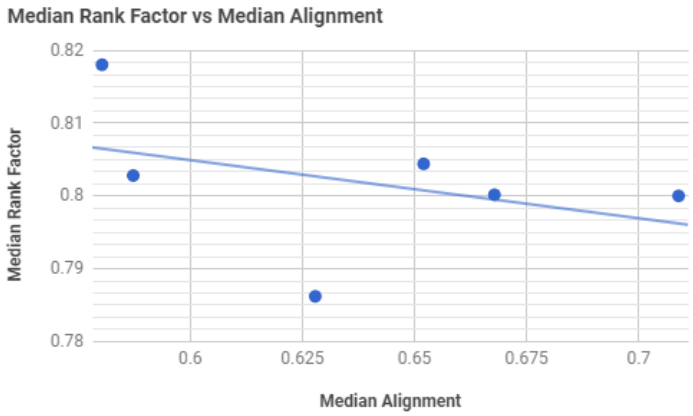


**Figure 23:** Charted relationship between median rank factor and median alignment

As can be seen there does appear to be a very weak negative correlation between the two variables which would contradict our hypothesis. The Pearson product-moment test correlation coefficient for this plot was -0.3855851179, which in turn gives a p-value of 0.451033 at n = 6. This p-value is obviously very high and well above the 0.05 and 0.10 confidence levels, meaning the null hypothesis that there is no correlation cannot be rejected.

This result of there being no statistically significant correlation is obviously not what was hypothesised would be the case, however it does allow for reflection on the co-occurrence matrix of the previous section. A positive correlation was hypothesised because

the supposed intrinsic susceptibility of the co-occurrence matrix model to common answer semantic elements (which would be reflected in the fragments) meant that a high alignment, caused by having common semantic properties with the answers of the nearest neighbours, would therefore also result in a high rank factor because semantically similar answers would be likely to share fragments resulting in high co-occurrence frequencies with the semantically similar question fragments.

Given that this does not appear to be the case it can be postulated that the co-occurrence matrix may not be as fragile to semantically common answers as previously thought. Another possibility is that the original hypothesis was flawed in that the median rank factors were calculated by excluding the given Prime Minister and the alignment was calculated by excluding all but the given Prime Minister so a high alignment meant that the Prime Minister was using similar answers to similar questions relative to themselves but the fragments used may not have been those used consistently by other Prime Ministers, thus leading to a case of high alignment but low rank factor.

***Alignment by Party***

In the introduction to this section the possibility of there being differences in alignment, then referred to as consistency, between the parties was mentioned because previous work cites a difference in ideological consistency by ideological group: *"Across diverse samples and attitude measures, liberals were typically higher in ideological consistency than conservatives"* [38]. The assumption within this analysis is that Labour typically embody liberal views more and the Conservatives as their name suggests embodying conservative views more.

To compare the alignments of the Labour and Conservative parties a mean alignment would be required for each as well as a standard deviation for each in order to do a comparison of two means test. It should be noted here the values output during the 'Median Alignment by Prime Minister' analysis were used for this analysis also. The mean alignment for each Prime Minister was output as one of the statistics during the previous analysis, giving us the table of mean alignments seen in figure 24 below.

It is also important to note at this point that because each Prime Minister's alignment was calculated relative only to themselves there is no bias in the dataset as previously mentioned in section one since although there are more Conservative Prime Ministers in the corpus this will have no effect on the Labour Prime Ministers alignments.

| Prime Minister | Mean Alignment |
|---|---|
| Thatcher | 0.7163819268 |
| Major | 0.6859041513 |
| Blair | 0.6266178483 |
| Brown | 0.5859407524 |
| Cameron | 0.6508160691 |
| May | 0.5810354799 |

**Figure 24:** Mean alignment by Prime Minister

Using the values of mean alignment for each Prime Minister, a party mean was calculated using the equation below in figure 25. This resulted in a party-mean alignment value of 0.6169541776 for Labour (from Blair and Brown) and 0.6896484953 for the Conservatives (from Thatcher, Major, Cameron and May). Given that alignment itself is a measure of consistency these values appear to contradict the hypothesis presented earlier since the Conservatives have a higher party level mean alignment than that of the Labour party as measured by our metrics.

$$\frac{\sum_{pm \in party} num\_pairs_{pm} * mean\_align_{pm}}{\sum_{pm \in party} num\_pairs_{pm}}$$

**Figure 25:** Party mean alignment formula

Now that party-mean alignment values were calculated the party standard deviation was calculated using the standard procedure where the party-mean was used as the mean and the individual Prime Ministers means were used as the values. This resulted in a party standard deviation of 0.03248413567 for Labour and 0.06839454392 for the Conservatives.

Having now calculated these figures a comparison of two means test was performed, whereby this would determine if there was a statistically significant difference in the means between the groups. Using these values, as well as n = 2 for the Labour party and n = 4 for the Conservatives a comparison of means test was conducted with the alternative hypothesis being that the values seen were statistically significantly different, this resulted in a p-value of 0.2435 which obviously exceeds both the 0.05 and 0.10 significance levels meaning the null hypothesis cannot be rejected and it cannot therefore be said there is a significant difference in the values calculated. This of course means that the results observed do not contradict those seen in the previously cited work on the topic of ideological consistency by ideological group.

## Model Evaluation & Possible Future Work

The system presented in this section is less technically complex than the system presented in section one to model a co-occurrence matrix, it is also more specific in its goal which was to allow a measure of consistency to be generated for question-answer pairs in a corpus. As such there are less improvements that can be made to this system in terms of its modelling unlike the co-occurrence matrix which could be expanded by adding new metrics and instead the improvements that can be made are in allowing the user greater control over the feature vector space and associated metrics.

The system uses a single feature vector space for the questions and answers, however it may be the case that a user wishes to use a vector space for the questions and a separate vector space for the answers. To this end a simple improvement to the system would be to introduce a 'split_space' flag whereby when set the system would create two vector spaces instead of one, perhaps with different parameters as specified by the user. Another possible improvement which could be made using a simple flag would be to allow the user to specify whether they want alignment distances normalised within the neighbours or within the entire vector space, this would allow the user to define for themselves how they wish to deal with the mathematical ambiguity discussed previously.

In the sphere of allowing the user greater control of the calculation of the alignment values another possibility would be to allow the user to pass define their own pair alignment calculation function and pass it to the system as a lambda function. This would allow the user to use their own notion of alignment and still result in obtaining a weighted average for each question-answer pair. A user defined alignment could for example consider whether the elements close in the vector space had similar emotive contents or referenced the same entities.

Considering allowing a user defined alignment measurement naturally opens the idea of allowing the user to also define a distance function allowing users to determine nearest neighbours in the vector space using alternatives to the currently used Euclidean metric. Another obvious possibility is to allow the user to pass a generic model, allowing for a plethora of different text representation models such as bag of words to be used. The only requirements for the generic model passed would be that the feature vectors produced are of fixed length and there is no duplication of feature vectors. This would also integrate symbiotically with the previous possibility of allowing the user to split the question and answer feature space into two distinct feature spaces by allowing one model to be used for question and one for answers if

different features should be weighted differently depending on whether the text is a question or answer.

One obvious issue with the current system is that the computation time for calculating the alignment values is high and grows at a rate of $O(n^2)$. To combat this the user could normalise distances within the neighbours using the previously mentioned flag as opposed to the entire vector space, however another possibility would be to implement as much of the system as possible to make calls to Cython functions which would be considerably faster than pure Python alone [46] [47]. In doing this, users could be empowered to compute alignments much quicker, reducing the currently large time investment needed to analyse a sizeable corpus.

Thus, the improvements that can be made to this system are not model based improvements like those that were discussed in section one for the fragment co-occurrence matrix and instead are based on improving the utility of the system to the end user, allowing them to modify the requisite components such as the underlying model or distance metrics in order to allow them to tailor the alignment metric to fit their needs. Indeed, the improvements listed here would make this system a much more universal solution for alignment problems whereas it currently fits a specific model and metric which were utilised for the research seen previously.

**Section Two Conclusion**

In this section the notion that a more consistent Prime Minister was superior at answering questions (measured using the rank factor of section one) was put to the test using data derived from a system combing Doc2Vec with alignment equations to generate a measure of consistency for each Prime Minister. The improvements that could be made to this system as discussed previously were not model improvements but instead accessibility improvements, allowing a greater scope of usability for alignment research.

Regarding the research conducted using this system no statistically significant evidence was found to support the hypothesis presented that a more consistent Prime Minister would be more adept at answering questions, however statistically significant evidence was found to suggest that a Prime Minister who answered more questions was more consistent. A theory presented by a previous work that people who were ideologically liberal were more consistent than those who were ideologically conservative [38] appeared to be refuted by the metrics generated in this work, but the values observed were found to be statistically insignificant and thus not sufficient to refute the previous work.

# Conclusion

Through the development of the software systems as seen in sections one and two a rich set of metrics was generated from a corpus comprised of ~215000 question-answer pairs from Prime Minister's Questions ranging from the Thatcher premiership to the early months of the May premiership. This set of metrics allowed trends in the strength and consistency of political answers to be analysed.

A general upward trend in the strength of political answers, as measured by the median rank factor, was observed with Tony Blair being the only outlier. Blair's rhetorical style may be the reason for his low median rank factor because as discussed there is previous work supporting the notion that he had a different style of speaking to other Prime Ministers [18]. A general downward trend was observed in the consistency, as measure by alignment, of political answers, with David Cameron lying outside the trend by being the only Prime Minister with an increase in consistency when compared to their predecessor.

No statistically significant correlation was found to support the hypothesis that a more consistent Prime Minister would be one who answers more strongly, however a statistically significant correlation was found between the number of answers given by a Prime Minister and their consistency implying a more experienced Prime Minister is a more consistent one.

Previous work suggested than liberals were ideologically more consistent than conservatives [38] and although our metrics appeared to contradict this suggestion the results were not statistically significant and thus the previous work cannot be refuted by the findings of this work. All the previously mentioned metrics and their analysis contribute significantly to the political analysis literature, particularly in the realm of UK Prime Ministers, validating the notion that Tony Blair had a different rhetorical for example, or adding a new point of view to the debate of ideological consistency.

Although each section had detailed discussions of possible improvements and future work to be undertaken there is one overarching piece of future work that would help validate the co-occurrence model of section one and any consistency metrics generated by the system of section two and that would be to apply the models to new datasets on which previous research has been conducted in order to compare and contrast the findings of this system to the findings of previous work.

# Bibliography


[1] A. Crowley, EEECS GitLab, 22 October 2017. [Online]. Available: https://gitlab.eeecs.qub.ac.uk/40121793/CSC3002_Computer_Science_Project.

[2] J. Zhang, A. Spirling and C. Danescu-Niculescu-Mizil, "Asking too much? The rhetorical role of questions in political discourse," in *EMNLP*, 2017.

[3] S. R. Bates, P. Kerr, C. Byrne and L. Stanley, "Questions to the Prime Minister: A Comparative Study of PMQs from Thatcher to Cameron," *Parliamentary Affairs,* vol. 67, no. 2, pp. 253-280, 2014.

[4] TotalPolitics, "30 facts about PMQs," TotalPolitics, 13 October 2010. [Online]. Available: https://www.totalpolitics.com/articles/news/30-facts-about-pmqs. [Accessed 2 February 2018].

[5] B. Allen, R. Fox, I. Geis-King, V. Gibbons, M. Korris, P. Pavlova and M. Raftery, "Tuned in or turned off? Public attitudes to Prime Minister's Questions," Hansard Society, 2014.

[6] G. Bush H W, Interviewee, *George H. W. Bush on Prime Minister's Questions.* [Interview]. 20 December 1991.

[7] P. Bull and P. Wells, "Adversarial Discourse in Prime Minister's Questions," *Journal of Language and Social Psychology,* vol. 31, no. 1, pp. 30-48, 2012.

[8] T. Holtgraves, "Language As Social Action: Social Psychology and Language Use," Lawrence Erlbaum Associates Inc, 2002, p. 38.

[9] J. D'Urso, "Corbyn v May in the battle to go viral at Prime Minister's Questions," BBC Political Research Unit, 13 December 2017. [Online]. Available: http://www.bbc.co.uk/news/uk-politics-42269771. [Accessed 20 February 2018].

[10] A. Therrien, "General election 2017: What caused Labour's youth vote surge?," BBC News Online, 16 June 2017. [Online]. Available: http://www.bbc.co.uk/news/uk-politics-40244905. [Accessed 20 February 2018].

[11] P. McAverty, I. Macleod, E. Tait, G. Baxter, A. Göker and M. Heron, "New Radicals? Digital Political Engagement in Post-Referendum Scotland. Interim Report.," Working Papers of the Communities & Culture Network+, 2015.

[12] J. Grimmer and B. M. Stewart, "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts," *Political Analysis,* vol. 21, no. 3, pp. 267-297, 2013.

[13] E. Goffman, "Replies and Responses," *Language in Society,* vol. 5, no. 3, pp. 257-313, 1976.

[14] J. Jeon, B. W. Croft and H. L. Joon, "Finding Semantically Similar Questions Based on Their Answers," in *SIGIR* , Salvador, Brazil, 2005.

[15] J. Leskovec, A. Rajaraman and J. D. Ullman, Mining of Massive Datasets, Cambridge University Press, 2014.

[16] Stanford University Natural Language Processing Group, "StanfordNLP CoreNLP GitHub," 22 January 2016. [Online]. Available: https://github.com/stanfordnlp/CoreNLP/blob/master/data/edu/stanford/nlp/patterns/surface/stopwords.txt. [Accessed 16 April 2018].

[17] Laerd Statistics, "Pearson Product-Moment Correlation," [Online]. Available: https://statistics.laerd.com/statistical-guides/pearson-correlation-coefficient-statistical-guide.php. [Accessed 29 April 2018].

[18] C.-B. Jonathan and L. Helms, "Comparative keyword analysis and leadership communication: Tony Blair - A study of rhetorical style," in *Comparative Political Leadership*, Basingstoke, Palgrave Macmillan, 2012, pp. 142-164.

[19] C. Prescott, "Internet access – households and individuals: 2017," Office for National Statistics , 3 August 2017. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/householdcharacteristics/homeinternetandsocialmediausage/bulletins/internetaccesshouseholdsandindividuals/2017. [Accessed 03 March 2018].

[20] S. Stieglitz and L. Dang-Xuan, "Emotions and Information Diffusion in Social Media - Sentiment of Microblogs and Sharing Behavior," *Journal of Management Information Systems,* vol. 29, no. 4, pp. 217-248, 2013.

[21] C. Tan, V. Niculae, C. Danescu-Niculescu-Mizil and L. Lee, "Winning Arguments: Interaction Dynamics and Persuasion Strategies in Good-faith Online Discussions," *Computing Research Repository,* vol. 1602.01103, 2016.

[22] M. Brysbaert, A. B. Warriner and K. Victor, "Concreteness ratings for 40 thousand generally known English word lemmas," *Behavior Research Methods,* vol. 46, no. 3, p. 904–911, 2013.

[23] S. M. Mohammed and P. D. Turney, "Crowdsourcing a Word–Emotion Association Lexicon," *Computational Intelligence,* vol. 29, no. 3, pp. 436-465, 2013.

[24] N. K. Steffens and S. A. Haslam, "Power through 'Us': Leaders' Use of We-Referencing Language Predicts Election Victory," *PLoS ONE,* vol. 8, no. 10, 2013.

[25] K. Searles and T. N. Ridout, "The Use and Consequences of Emotions in Politics," Emotion Rsearcher, February 2017. [Online]. Available: http://emotionresearcher.com/the-use-and-consequences-of-emotions-in-politics/. [Accessed 24 February 2017].

[26] P. E. Jones, L. H. Hoffman and D. G. Young, "Online emotional appeals and political participation: The effect of candidate affect on mass behavior," *New Media & Society,* vol. 15, no. 7, p. 1132–1150, 2012.

[27] B. Cammaerts, “The strategic use of metaphors by political and media elites: the 2007-11 Belgian constitutional crisis,” *International Journal of Media & Cultural Politics,* vol. 8, no. 2-3, pp. 229-249, 2012.

[28] A. M. Durik, M. A. Britt, R. Reynolds and J. Storey, “The Effects of Hedges in Persuasive Arguments: A Nuanced Analysis of Language,” *Journal of Language and Social Psychology,* vol. 27, no. 3, pp. 217-234, 2008.

[29] L. Wan, N. Beauchamp, S. Shugars and K. Qin, “Winning on the Merits: The Joint Effects of Content and Style on Debate Outcomes,” *Computing Research Repository,* vol. 1705.05040, 2017.

[30] V. Prabhakaran, A. Arora and O. Rambow, “Staying on Topic: An Indicator of Power in Political Debates,” in *Empirical Methods for Natural Language Processing*, 2014.

[31] A. Boydstun, M. Pietryka, R. A. Glazier and P. Resnik, “Real-Time Reactions to a 2012 Presidential Debate: A Method for Understanding Which Messages Matter,” *Public Opinion Quarterly,* vol. 78, no. 1, pp. 330-343, 2014.

[32] R. B. Cialdini, Influence: The Psychology of Persuasion, HarperBusiness, 2006.

[33] K. Kaur and V. Gupta, “A Survey of Topic Tracking Techniques,” *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 2, no. 5, 2012.

[34] K. Rajaraman and A.-H. Tan, “Topic Detection, Tracking, and Trend Analysis Using Self-Organizing Neural Networks (PAKDD),” in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Hong Kong, 2001.

[35] C. Clifton, R. Cooley and J. Rennie, “TopCat: Data Mining for Topic Identification in a text corpus,” *IEEE Transactions on Knowledge and Data Engineering,* vol. 16, no. 8, pp. 949 - 964, 2004.

[36] T. A. Kreps, A. C. Merritt and K. Laurin, “Hypocritical Flip-Flop, or Courageous Evolution? When Leaders Change Their Moral Minds,” *Journal of Personality and Social Psychology,* vol. 113, no. 5, pp. 730-752, 2017.

[37] E. Oldfield, “Party leaders 'talking God' more,” 11 August 2011. [Online]. Available: https://www.theosthinktank.co.uk/comment/2008/09/14/party-leaders-talking-god-more. [Accessed 26 March 2018].

[38] P. Kesebir, E. Phillips, J. Anson, T. Pyszczynski and M. Motyl, “Ideological Consistency Across the Political Spectrum: Liberals are More Consistent But Conservatives Become More Consistent When Coping with Existential Threat,” *SSRN Electronic Journal,* vol. 10.2139/ssrn.2215306, 2013.

[39] Q. Le and T. Mikolov, “Distributed Representations of Sentences and Documents,” *Computing Research Repository,* vol. 1405.4053, 2014.

[40] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” *Computing Research Repository,* vol. 1301.3781, 2013.

[41] S. Massie, S. Craw, A. Donati and N. Wiratunga, “From Anomaly Reports to Cases,” in *International Conference on Case-Based Reasoning*, Belfast, 2007.

[42] M. A. Raghunandan, N. Wiratunga, S. Massie, S. Chakraborti and D. Khemani, “Evaluation Measure for TCBR Systems,” in *9th European Conference on Case-Based Reasoning*, Trier, 2008.

[43] R. Řehůřek, “Models.doc2vec – Deep learning with paragraph2vec,” [Online]. Available: https://radimrehurek.com/gensim/models/doc2vec.html. [Accessed 29 April 2018].

[44] Python Software Foundation, “Multiprocessing — Process-based parallelism,” [Online]. Available: https://docs.python.org/3/library/multiprocessing.html. [Accessed 29 April 2018].

[45] Python Software Foundation, “Types — Dynamic type creation and names for built-in types,” [Online]. Available: https://docs.python.org/3/library/types.html#types.SimpleNamespace. [Accessed 29 April 2018].

[46] J. F. Puget, “A Speed Comparison Of C, Julia, Python, Numba, and Cython on LU Factorization,” IBM Developer Works, 15 January 2016. [Online]. Available: https://www.ibm.com/developerworks/community/blogs/jfp/entry/A_Comparison_Of_C_Julia_Python_Numba_Cython_Scipy_and_BLAS_on_LU_Factorization?lang=en. [Accessed 10 April 2018].

[47] T. Craven, “Faster Python with Cython and PyPy: Part 2,” Cardinal Peak, 19 August 2016. [Online]. Available: https://cardinalpeak.com/blog/faster-python-with-cython-and-pypy-part-2/. [Accessed 10 April 2018].