UNIVERSITY OF BIRMINGHAM

School of Computer Science
College of Engineering and Physical Sciences

MSc. Project

Analysing news articles about Russia's war on Ukraine using Latent Dirichlet Allocation based topic modelling

---

Submitted in conformity with the requirements
for the degree of MSc. Artificial Intelligence and Machine Learning
School of Computer Science
University of Birmingham

Nelson Quintanilla Castro
Student ID: 2291960
Supervisor: Dr Mohammed Bahja

September 2022

Nelson Quintanilla Castro

MSc. Project
Analysing news articles about Russia's war on Ukraine using Latent Dirichlet Allocation based topic modelling

Nelson Quintanilla Castro

Contents

Table of Abbreviations

- LDA – Latent Dirichlet Allocation
- API – Application Programming Interface
- NLTK – Natural Language Toolkit
- BOW – Bag of Words

List of Figures

List of Tables

1 Overview

1.1 Abstract

Topic Modelling is one of the most effective techniques used in the field of natural language processing for the discovery of latent information within a given corpus. The main focus of this work is aimed at using Latent Dirichlet Allocation, which is one of the most popular topic modelling methods.

The purpose of this work is to use an LDA based topic modelling approach to capture the main themes or topics that can be found on news articles from The Guardian digital newspaper that are exclusively related to the ongoing Russian invasion of Ukraine. Furthermore, an interpretation of the topics obtained and its latent significance for the current state of the situation and its future is desired.

We would be using Gensim, which is a well known Python library that provides fast, efficient, and a scalable implementation of the LDA algorithm. In order to train our model first we are going to extract our data, and manipulate it in a pre-processing step needed before feeding the data to our model. Then we are going to train multiple models with different values of number of topics and further parameter tuning. Subsequently, we will evaluate and decide what model is best for our purpose and we will interpret the results.

This work shows that: preprocessing is a fundamental step when implementing LDA based topic modelling to obtain good results; considerably large dataset also leads to good results; the process of tuning the parameters of our model could get computationally demanding if the number of topics chosen is too large; and lastly, LDA is a great tool to analyse large datasets and find the main topics contained within them. Furthermore, it is possible to give an interpretation of the obtained topics and its related words with the highest probabilities.

1.2 Acknowledgements

This work could not have been possible without the academic support from Dr Mohammed Bahja and Dr Felipe Orihuela-Espina.

Dr Mohammed Bahja always motivated me and academically advised me how to approach many challenges that I faced during the dissertation process.

Dr Felipe Orihuela, who also was my tutor during the academic year of my Master's degree, was fundamental for me to get a better sense of how the UK education system works and how to better approach challenges but most importantly how to always strive for excellence even during hard times or in adversity. During the dissertation process his advice was invaluable and has only enriched my academic experience.

2 Introduction

The ongoing conflict between Russia and Ukraine is a topic of pivotal importance not only for the protagonist's regions but also for the whole world. This war is already causing many negative economic, political, and social consequences on a different scale for many countries (Caldara et al. 2022). As a result, this has become of paramount concern for governments and, subsequently for all types of media.

This is why there is currently a tremendous amount of information that has been generated about the Russo-Ukrainian conflict and there is no systematic way to analyse this corpora to examine what is currently happening regarding this conflict, to discover the main themes contained within it, and possibly draw conclusions about it.

At the same time, it is not humanly possible to read and study large quantities of articles with this thematic in a short period of time. To do this, experts in machine learning have created probabilistic topic modelling, a collection of several algorithms which are used to identify and interpret sizeable sets of documents that contain topical information. We can use topic modelling to identify the primary ideas that run through a sizable and otherwise unstructured collection of papers. We can arrange the collection in accordance with the themes found by applying topic modelling (Blei 2012).

For the purpose of this work, latent dirichlet allocation (LDA) is going to be the topic modelling algorithm of choice. A collection of texts can be broken down into their key topics using LDA, where a topic is defined as a probability distribution over a vocabulary (Blei n.d.).

A set number of topics are proposed by LDA, and it is assumed that each document in a collection of documents reflects a combination of those themes. Under these presumptions, probabilistic inference techniques identify an embedded theme structure in a document collection. LDA offers a technique to easily summarise, peruse, and search huge document collections with this structure (Blei et al. 2010).

The main motivation of this work is to capture the main themes or topics that can be found on news articles available on the internet that cover events related to Rusia's war on Ukraine. This represents an important topic to study because there is a need to derive insightful information from this vast amount of news about the war that are being generated in mass all over the world.

Furthermore, it is of utter importance to observe and detect significant patterns that are being communicated to the general public through news articles so we can predict consequences of the current state of the situation on the short term such as the economical, political, and social impact on human lives.

In this paper we provide an analysis of news articles retrieved from The Guardian newspaper that cover events related to the Russian invasion of Ukraine by utilising LDA based topic modelling with the purpose of obtaining abstract

topics within a corpora of news articles. We also offer a way to assess the effectiveness and outcomes of our model. Additionally, we present the validation of our model by comparing it against human judgement. In order to allow others to duplicate our findings, we have also made our data sets publicly available. Further to this, we use pyLDAvis to interpret the topics obtained by our model in a more comprehensible and visual way. This software offers a global view of the topics and how they differ from one another while also enabling a close examination of the terms most closely associated with each specific topic (Sievert & Shirley 2014).

The structure of this paper is as follows. In section 3 we examine the most recent studies in this field. In Section 4 we present the specifics of our approaches, including a description of data extraction, preprocessing, parameter tuning of our model, and evaluation metrics implemented. We provide our findings and engage in a discussion in Section 5. The discussion is presented in part 6, and our conclusion is presented in section 7.

3 Literature Review

3.1 Subsection

Many researchers have used LDA based topic modelling in order to analyse datasets with different tematics.

3.2 Background

3.2.1 Topic Modelling

Topic models are computer algorithms that uncover hidden patterns of word occurrence by analysing the distribution of words in a corpus of documents. The output is a list of themes composed of clusters of words that co-occur in these documents in accordance with specific patterns (Jacobi et al. 2016).

3.2.2 Latent Dirichlet Allocation

Latent Dirichlet Allocation is a hierarchical probabilistic model used to divide a collection of documents into their salient topics, where a "topic" for LDA is a probability distribution over a vocabulary (Blei 2012).

The Figure 1 represents the intuition behind the LDA algorithm. We suppose that some number of "themes," or word distributions, exist for the entire corpus (far left). It is believed that each document is generated as follows. First, select a distribution over the subjects (the histogram on the right). Then, for each word, select a topic assignment (the coloured coins) and the matching topic word. These subjects and topic allocations are illustrative; they are not based on actual data (Blei 2012).

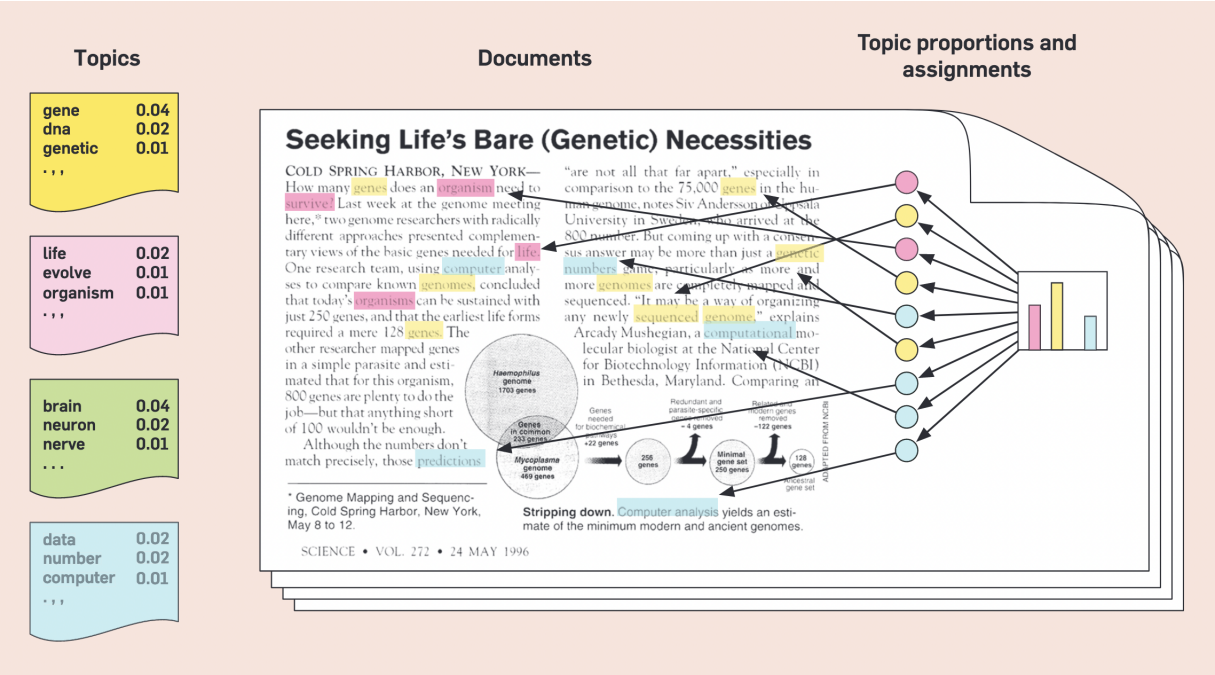

Figure 1: *Intuition behind LDA. Taken from (Blei 2012).*

4 Methodology

In this section we explain the steps we took in order to build and prepare our dataset, the model generation and parameter tuning process, methods of evaluation for our model, and lastly visualisation and interpretation of our resulting topics. The Figure 2 summarises the method we just mentioned.

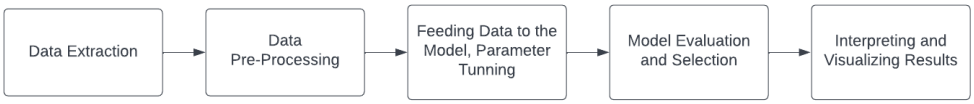

Figure 2: *Diagram that summarises steps taken during the development of our work.*

4.1 Data Extraction

To obtain a significantly large dataset of reliable news articles related to the Russian invasion of Ukraine we decided to use The Guardian digital newspaper. The reasoning behind this decision is because The Guardian is one of the most trusted newspapers in the UK and Europe in terms of objective journalism (Matsa et al. 2018). Additionally, they provide The Guardian – Open Platform, which is their public web service or application programming interface (API) that provides free access to all of the content that the Guardian produces, organised by tags and sections (*theguardian / open platform - documentation / overview* n.d.). This is specially useful in our case because we need to filter the news to reduce noise in our data and have articles relevant to the ongoing conflict between Russia and Ukraine.

To narrow-down the retrieval of the articles, we requested published data between the 2nd of February of 2022 (day in which Russia invaded Ukraine) and the 31st of August of 2022. Additionally, we created a function to interact with the API and retrieve news articles filtered by search term and section. This is to avoid obtaining non relevant news; for instance, we used the section "world" to point to data classified in this category and not other categories such as "sports" or "lifestyle". Furthermore, we used 2 search terms: Ukraine and Russia. We obtained 2 different datasets corresponding to the mentioned search terms. After that, we merged both datasets and removed duplicates that could have existed in both to avoid redundancy. We also removed non pertinent articles that contained information about topics not related the conflict.

The resulting dataset is a json containing a list of 2275 news articles with their id, title of the article, body of the article, and url to read the article on the web. Such dataset is available in the repository included in Section 8 so it can be used for future research or work.

4.2 Data Pre-Processing

Before feeding the data to our model it is required to prepare the data (Jacobi et al. 2016).

Following is a detailed description of the processes we took when completing the analysis for our case study.

4.2.1 Dataset Relevant Features

Before starting pre-processing the data we display important features from our dataset, such as the number of articles to be used and the average word count per article. This is just to have a rough idea of the length of each document we are using for the purposes of training our model. In our case, we have a total of 2275 news articles and an average of 864 words per article.

4.2.2 Normalisation and Tokenisation

A topic model does not analyse documents directly; rather, it employs a so-called document–term matrix that is derived from the documents. This matrix displays the frequency for each term (word) in each document (Jacobi et al. 2016). The initial step in the creation of this matrix is normalisation and tokenisation, for which each page is converted into a list of lowercase words, ignoring short and long words.

4.2.3 Stop Word Removal

Stop words are the words in a stop list that are filtered out prior to processing natural language data (text) due to their insignificance (Rajaraman & Ullman 2011).

We compared three standard lists of stop words from extensively used libraries for machine learning applications: Natural Language Toolkit (NLTK) (Bird n.d.), scikit-learn (Pedregosa et al. n.d.), and spaCy (Honnibal & Montani 2017). There is a file dedicated to do this comparison on our repository included in Section 8. However, it is worth mentioning that our final decision was to merge the lists of stop words since this leads to better results in terms of removing words that are not adding any meaning, for instance "but", "end", and "the".

4.2.4 Lemmatisation

Lemmatisation is an important part of the preprocessing of the data. It reduces all words to their "lemma" by the use of a lexicon and regular conjugation rules. Thus, lemmatization reduces "is" and "were" to their respective lemma (to be) "be" (Jacobi et al. 2016). There is also another technique called stemming which removes the endings of the words leaving just the "stems". Using the commonly employed Porter stemming algorithm, for instance, "weaknesses" becomes "weak" while "failures" and "failure" both become "failur" (Jacobi et al. 2016).

In our case lemmatisation was preferred over stemming because it produced better results as compared to stemming which sometimes returned words that were missing letters at the end and therefore did not exist in the English dictionary.

For the analysis presented here we used the lemmatiser from spaCy (Honnibal & Montani 2017) which which assigns tokens' base forms using criteria derived from part-of-speech tags (POS).

4.2.5 Transforming the documents in a vectorised form (Bag of Words Representation)

As we mentioned earlier, before we feed the data to the model we need to transform the documents into a document-term matrix. This way of expressing our dataset is known as Bag of Words (BOW) representation. BOW represents the content of the documents by showing the frequency of the word on every document. Once we have this, we are ready to start training our model.

After using the aforementioned strategies, we have removed a number of terms from our initial tokenised dataset and adjusted the remaining words' structure to make them more effective for training our model. In the following section, we are going to delve into the model training and tunning.

4.3 Model Training and Parameter Tunning

There are numerous tools available for Topic modelling and analysis, including professional and amateur software, commercial and open-source software. Some examples of these well known tools that have been used by a large number of scholars in the past are Mallet (McCallum 2002), TMT (Ramage et al. 2009), and Gensim (Jelodar et al. 2019).

For the training of our model we used Gensim, which is a Python package that provides algorithms for Latent Dirichlet Allocation that are memory-efficient, scalable, and quick (Rehurek & Sojka 2010, Řehůřek & Sojka n.d.).

In order to train our LDA model we need to provide it with different parameters as it can be observed in Figure 3. From all the parameters, one of the most important ones is the number of topics we want to extract from our dataset as this one affects greatly on the quality of the results. Furthermore, choosing the number of topics in LDA is typically determined by assessing the perplexity of numerous models built with varying values of number of topics as a parameter and choosing the one with the lowest perplexity value (Blei et al. 2010).

Perplexity is a metric for evaluating language models, where a low score denotes a model with superior generalisation, as determined by (Asmussen & Møller 2019, Blei & Lafferty 2007, Xu & Raschid 2016, Zhao et al. 2015).

Reducing the perplexity score is equivalent to increasing the chance of all articles belonging to a topic. The criteria for picking the optimal number of topics is to strike a balance between a manageable number of topics and the lowest possible level of perplexity. The optimal amount of topics can vary substantially depending on the purpose of the analysis. As a general guideline, fewer topics are

utilised for a general overview, while more topics are used for a more detailed look (Asmussen & Møller 2019). In our case, we geared our experimentation towards obtaining a general overview.

Coherence score is also extensively used in topic model experiments as an assessment tool (Albalawi et al. 2020, Ferner et al. 2020, Li et al. 2019, Towne et al. 2016) and it indicates the degree of semantic similarity of high-scoring terms in the text and helps to differentiate the semantic interpretation of themes based on statistical inference (Ray et al. 2019).

Having mentioned that, once we generated our corpus or Bag of Words representation of our dataset and our dictionary (a dictionary object that contains every single word on the entire dataset without duplicates), we created a function to generate many models at once with a different value of number of topics.

The first run we did was for models with 2, 8, 14, 20, 26, 32, and 38 number of topics. As we will explain further, we did several runs with different number of topics. At this point we computed the coherence score and perplexity for each of the models generated and as it can be observed in figure **??**. We assessed the calculated perplexity and coherence and decided to do a second run for models with 2, 4, 6, 8, 10, 12, and 14 number of topics.

We assessed the coherence and perplexity values, as well as the associated words with the highest probability for each of the topics of the generated models as it can be seen on Figure 4, and based on that we started adding custom words to our list of stop words. The purpose of this is to avoid getting some of the words repeated in many topics (this is the case for words that occur in the majority of the documents). Therefore, we removed the following words: Ukraine, Ukrainian, Russia, Russian; and ran our model a third time.

We assessed again the metrics in addition to the associated words for each of the topics of the model generated and this time we removed the following words: war, people, city; and ran our model a fourth time.

Lastly, we assessed in a similar manner, removed the words: country, year, day; and ran our model a fifth time.

For each of the runs in the process previously described we also did several other runs to tune the remaining parameters of the model by trial and error. When the results were better, we continued with the next iteration of runs. In the end, we selected the 3 best models from the last run. From these 3 models, we are going to find out which one is the best one utilising human judgement metrics.

4.4 Model Evaluation and Selection

To further evaluate our models we used human judgement metrics as well. We employ two tasks to build a formal environment in which humans can evaluate the two components of a topic model's latent space. The first element is the composition of the topics. We construct a task to determine whether a topic

```
lda_model_ = gensim.models.ldamulticore.LdaMulticore(
    corpus=corpus_,
    num_topics=num_topics_,
    id2word=id2word_,
    workers=None,
    chunksize=chunksize,
    passes=passes,
    alpha=alpha,
    eta=eta,
    decay=decay,
    offset=offset,
    eval_every=eval_every,
    iterations=iterations,
    gamma_threshold=gamma_threshold,
    minimum_probability=minimum_probability,
    random_state=random_state,
    minimum_phi_value=minimum_phi_value,
    per_word_topics=per_word_topics,
    dtype=dtype
)
```

Figure 3: *LdaMulticore function from Gensim to train or LDA model.*

has semantic coherence that is human-identifiable. This activity is known as word intrusion, as participants must recognise a fictitious word that has been put into a topic. The second assignment evaluates the validity of the relationship between a document and a topic. This task is known as topic incursion because the subject must identify a topic that the model did not correlate with the document (Nikolenko et al. 2017).

In Figure 5 it can be observed the human tasks we provided for 10 users to answer. The purpose of these tasks was to select the best model amongst the three best models we picked in the last step. For each model, in total, each user responded to 5 word intrusion challenges and 3 topic intrusion challenges. The url to the questionary can be found in the repository included in Section 8. Responses are no longer accepted but you can see the results.

To create the questionnaire, we used the 3 models in 3 news articles. For the word intrusion task, six words are presented to the individual in a random order. The user's objective is to identify the word that does not belong with the others, also known as the intruder. Five of the six words correspond to the highest probable words from a topic but the sixth word is an intruder (Nikolenko et al. 2017)..

The topic intrusion challenge examines if a topic model's breakdown of documents into a variety of subjects corresponds to human evaluations of the document's content (Nikolenko et al. 2017).

Both experiments were made in a similar fashion to (Chang et al. n.d., Nikolenko et al. 2017). Word and topic intruders were selected at random from terms and topics not present in the topic and text, respectively, similar to (Nikolenko et al.

```
Topics for a model trained for 6 number of topics
[(0,
  '0.009*"uk" + 0.008*"government" + 0.006*"sanction" + 0.006*"gas" + '
  '0.005*"company" + 0.004*"home" + 0.004*"refugee" + 0.004*"include" + '
  '0.004*"family" + 0.004*"new"'),
 (1,
  '0.010*"putin" + 0.007*"president" + 0.006*"military" + 0.006*"minister" + '
  '0.006*"eu" + 0.006*"nato" + 0.005*"force" + 0.005*"sanction" + '
  '0.005*"invasion" + 0.005*"zelenskiy"'),
 (2,
  '0.008*"macron" + 0.006*"le" + 0.006*"price" + 0.006*"pen" + 0.006*"party" + '
  '0.006*"food" + 0.005*"election" + 0.005*"vote" + 0.005*"france" + '
  '0.005*"right"'),
 (3,
  '0.012*"force" + 0.009*"kyiv" + 0.006*"civilian" + 0.006*"military" + '
  '0.006*"mariupol" + 0.006*"attack" + 0.005*"soldier" + 0.005*"kill" + '
  '0.005*"region" + 0.004*"leave"'),
 (4,
  '0.004*"time" + 0.004*"work" + 0.004*"like" + 0.004*"know" + 0.004*"tell" + '
  '0.004*"putin" + 0.003*"family" + 0.003*"man" + 0.003*"come" + 0.003*"want"'),
 (5,
  '0.006*"israel" + 0.005*"world" + 0.005*"israeli" + 0.004*"soviet" + '
  '0.003*"medvedchuk" + 0.003*"medvedev" + 0.003*"nazi" + 0.003*"palestinian" '
  '+ 0.002*"abortion" + 0.002*"jewish"')]
```

Figure 4: *Visualising topics and their 10 highest probable keywords for a model generated with 6 number of topics*

2017).

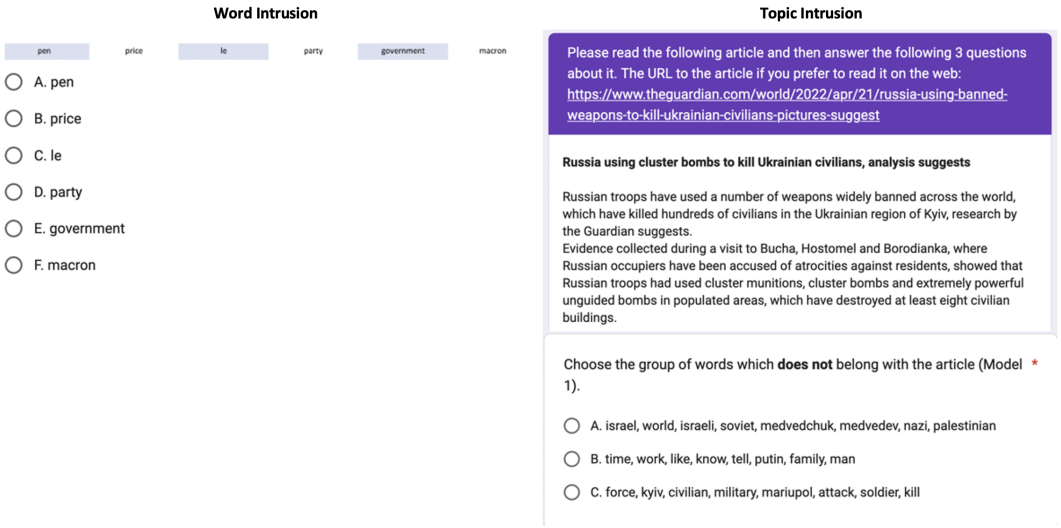

Figure 5: *Word intrusion task (left): the users are asked to select what of the words do not belong with the rest of the words. Topic intrusion task (right): the users are asked to read a news article and later is presented with 3 topics represented with their highest probable words, of which 1 of the topics does not belong to the document.*

5 Results

In Figure 6, Figure 7, and Figure 8 we can see the topics for each of the 3 models we were comparing. Each topic is represented for the 10 words with the highest probability.

In Figure 9 we can observe the plots of number of topics against coherence and against perplexity.

In the Table 1 we have the results of the experiments corresponding to the human judgement tasks.

Experiment	Model	Correct Answers	Total Answers	Ratio
Word intrusion	Model 1	13	50	0.260
	Model 2	10	50	0.200
	Model 3	14	50	0.280
Topic Intrusion	Model 1	16	30	0.533
	Model 2	17	30	0.567
	Model 3	13	30	0.433

Table 1: *Results: word intrusion and topic intrusion experiments*

```
Topics for a model trained for 6 number of topics
[(0,
  '0.009*"uk" + 0.008*"government" + 0.006*"sanction" + 0.006*"gas" + '
  '0.005*"company" + 0.004*"home" + 0.004*"refugee" + 0.004*"include" + '
  '0.004*"family" + 0.004*"new"'),
 (1,
  '0.010*"putin" + 0.007*"president" + 0.006*"military" + 0.006*"minister" + '
  '0.006*"eu" + 0.006*"nato" + 0.005*"force" + 0.005*"sanction" + '
  '0.005*"invasion" + 0.005*"zelenskiy"'),
 (2,
  '0.008*"macron" + 0.006*"le" + 0.006*"price" + 0.006*"pen" + 0.006*"party" + '
  '0.006*"food" + 0.005*"election" + 0.005*"vote" + 0.005*"france" + '
  '0.005*"right"'),
 (3,
  '0.012*"force" + 0.009*"kyiv" + 0.006*"civilian" + 0.006*"military" + '
  '0.006*"mariupol" + 0.006*"attack" + 0.005*"soldier" + 0.005*"kill" + '
  '0.005*"region" + 0.004*"leave"'),
 (4,
  '0.004*"time" + 0.004*"work" + 0.004*"like" + 0.004*"know" + 0.004*"tell" + '
  '0.004*"putin" + 0.003*"family" + 0.003*"man" + 0.003*"come" + 0.003*"want"'),
 (5,
  '0.006*"israel" + 0.005*"world" + 0.005*"israeli" + 0.004*"soviet" + '
  '0.003*"medvedchuk" + 0.003*"medvedev" + 0.003*"nazi" + 0.003*"palestinian" '
  '+ 0.002*"abortion" + 0.002*"jewish"')]
```

Figure 6: *Model with 6 number of topics from run 5. Each topic contains the 10 words with the highest probability within that topic.*

It can be observed that the word intrusion metric was not very decisive amongst

```
Topics for a model trained for 8 number of topics
[(0,
  '0.010*"government" + 0.009*"uk" + 0.009*"gas" + 0.006*"company" + '
  '0.006*"refugee" + 0.005*"energy" + 0.005*"home" + 0.005*"visa" + '
  '0.005*"family" + 0.004*"germany"'),
 (1,
  '0.008*"putin" + 0.008*"military" + 0.007*"force" + 0.007*"president" + '
  '0.006*"nato" + 0.006*"minister" + 0.006*"zelenskiy" + 0.005*"moscow" + '
  '0.005*"defence" + 0.005*"invasion"'),
 (2,
  '0.010*"macron" + 0.008*"le" + 0.008*"price" + 0.007*"pen" + 0.007*"food" + '
  '0.006*"france" + 0.006*"grain" + 0.005*"vote" + 0.005*"french" + '
  '0.005*"port"'),
 (3,
  '0.011*"force" + 0.009*"kyiv" + 0.007*"civilian" + 0.006*"mariupol" + '
  '0.006*"soldier" + 0.005*"attack" + 0.005*"kill" + 0.005*"military" + '
  '0.004*"leave" + 0.004*"region"'),
 (4,
  '0.005*"tell" + 0.005*"work" + 0.004*"family" + 0.004*"time" + 0.004*"know" '
  '+ 0.004*"leave" + 0.004*"like" + 0.004*"man" + 0.003*"news" + '
  '0.003*"group"'),
 (5,
  '0.005*"medvedchuk" + 0.005*"medvedev" + 0.005*"african" + 0.004*"world" + '
  '0.003*"china" + 0.003*"africa" + 0.002*"trophy" + 0.002*"road" + '
  '0.002*"schevchenko" + 0.002*"time"'),
 (6,
  '0.009*"putin" + 0.005*"soviet" + 0.005*"world" + 0.004*"like" + '
  '0.004*"time" + 0.004*"think" + 0.003*"right" + 0.003*"crime" + 0.003*"new" '
  '+ 0.003*"gorbachev"'),
 (7,
  '0.011*"eu" + 0.010*"sanction" + 0.009*"putin" + 0.006*"government" + '
  '0.005*"uk" + 0.005*"european" + 0.004*"abramovich" + 0.004*"minister" + '
  '0.004*"include" + 0.003*"party"')]
```

Figure 7: *Model with 8 number of topics from run 5. Each topic contains the 10 words with the highest probability within that topic.*

the three models. All of the models scored low. On the other hand, Model 2 and Model 1 have a score significantly higher than the score for Model 3.

Taking into account the results from the human judgement task in addition to the analysis given by the coherence score and perplexity of our models, it can be concluded that the best model for the purposes of obtaining a general overview of our dataset through the topics is Model 2 which corresponds to a model with 8 number of topics.

In the following section we are going to discuss further the implications of selecting this model and how we can interpret the topics and what do they tell us about the current situation between Russia and Ukrania.

Additionally, to visualise the topics in an interactive way, we used pyLDAvis which is a Python library for the interactive display of topic models (Sievert & Shirley 2014). In the Figure 10 you can observe the visualisation of the topics for the model 2 with 8 number of topics.

```
Topics for a model trained for 10 number of topics
[(0,
  '0.012*"gas" + 0.010*"company" + 0.008*"government" + 0.007*"sanction" + '
  '0.007*"energy" + 0.005*"germany" + 0.005*"include" + 0.005*"state" + '
  '0.004*"business" + 0.004*"supply"'),
 (1,
  '0.009*"force" + 0.009*"military" + 0.007*"putin" + 0.007*"president" + '
  '0.006*"zelenskiy" + 0.006*"nato" + 0.006*"defence" + 0.006*"moscow" + '
  '0.006*"minister" + 0.005*"invasion"'),
 (2,
  '0.011*"price" + 0.010*"food" + 0.010*"grain" + 0.009*"port" + 0.008*"world" '
  '+ 0.007*"export" + 0.006*"global" + 0.006*"oil" + 0.005*"crisis" + '
  '0.005*"sea"'),
 (3,
  '0.010*"force" + 0.009*"kyiv" + 0.006*"civilian" + 0.006*"soldier" + '
  '0.006*"mariupol" + 0.005*"kill" + 0.005*"leave" + 0.005*"attack" + '
  '0.004*"military" + 0.004*"child"'),
 (4,
  '0.005*"tell" + 0.004*"time" + 0.004*"man" + 0.004*"military" + 0.004*"work" '
  '+ 0.004*"family" + 0.004*"know" + 0.003*"group" + 0.003*"soldier" + '
  '0.003*"official"'),
 (5,
  '0.006*"medvedchuk" + 0.006*"african" + 0.004*"world" + 0.003*"africa" + '
  '0.003*"western" + 0.003*"china" + 0.003*"group" + 0.002*"time" + '
  '0.002*"soviet" + 0.002*"road"'),
 (6,
  '0.010*"putin" + 0.007*"soviet" + 0.006*"world" + 0.005*"crime" + '
  '0.004*"gorbachev" + 0.004*"court" + 0.004*"international" + '
  '0.003*"president" + 0.003*"state" + 0.003*"time"'),
 (7,
  '0.014*"eu" + 0.010*"putin" + 0.010*"sanction" + 0.006*"european" + '
  '0.006*"government" + 0.005*"uk" + 0.004*"minister" + 0.004*"germany" + '
  '0.004*"leader" + 0.004*"president"'),
 (8,
  '0.007*"uk" + 0.007*"government" + 0.006*"refugee" + 0.006*"home" + '
  '0.006*"macron" + 0.005*"family" + 0.005*"right" + 0.005*"china" + '
  '0.004*"visa" + 0.004*"new"'),
 (9,
  '0.010*"putin" + 0.008*"protest" + 0.006*"opposition" + 0.006*"medium" + '
  '0.006*"orban" + 0.005*"channel" + 0.005*"kremlin" + 0.005*"moscow" + '
  '0.005*"invasion" + 0.004*"russians"')]
```

Figure 8: *Model with 10 number of topics from run 5. Each topic contains the 10 words with the highest probability within that topic.*

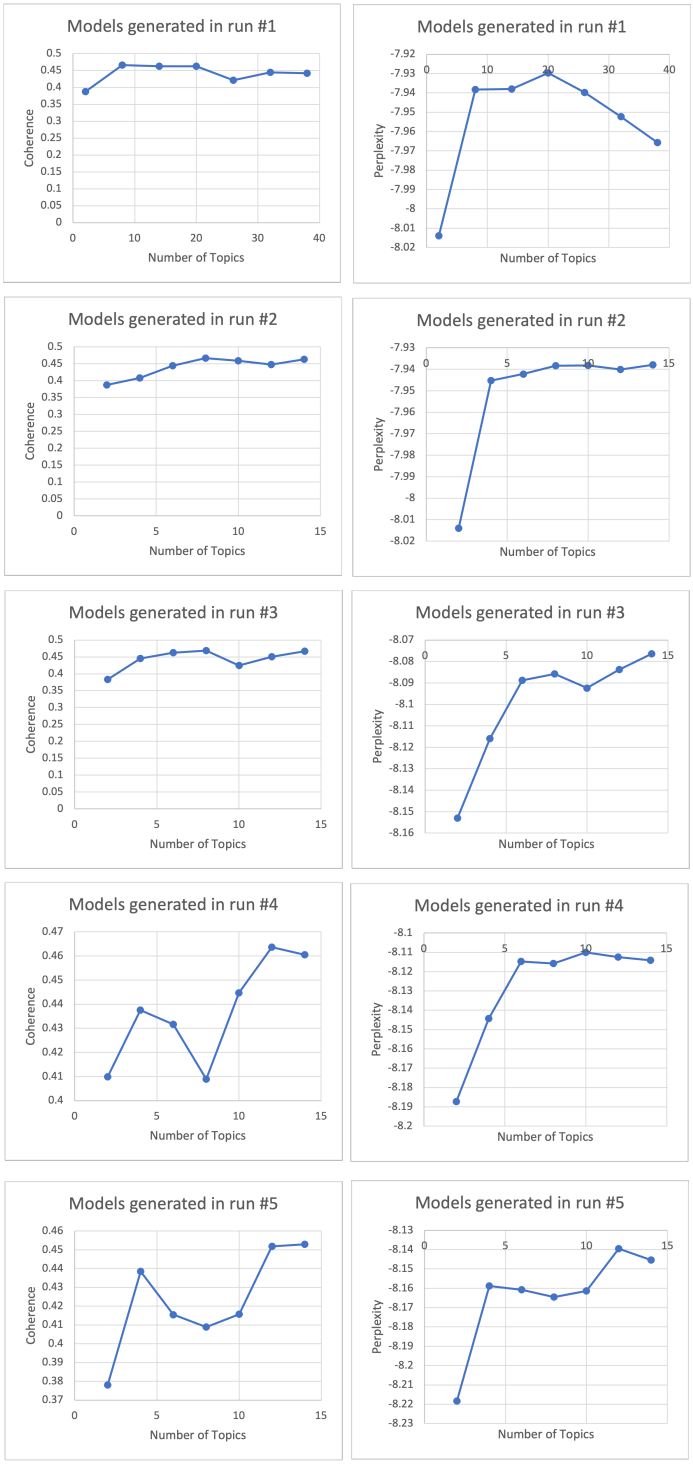

Figure 9: *Plots of coherence and perplexity for the models generated on each run during the parameter tuning process.*

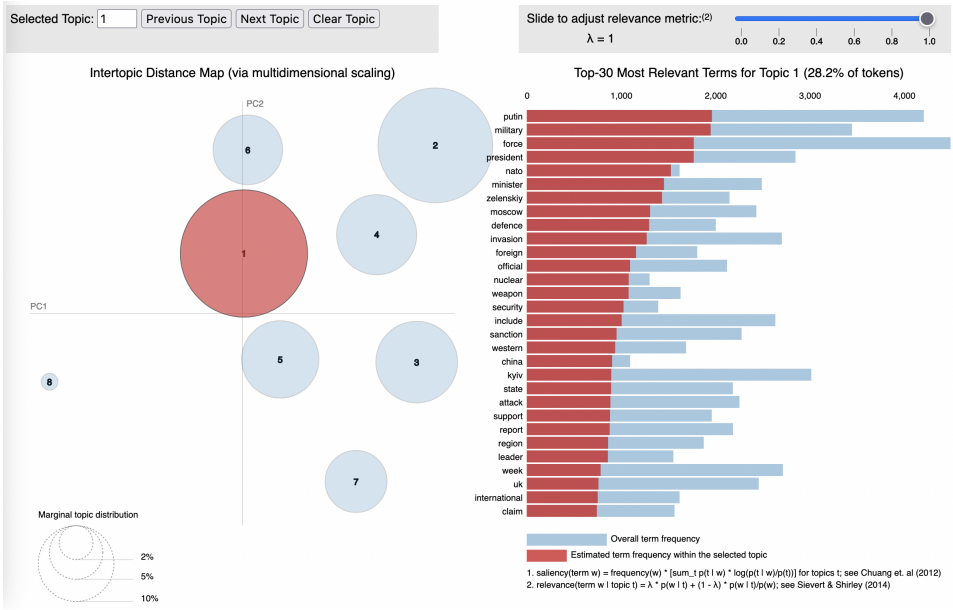

Figure 10: *Visualisation of the topics for the model 2 which corresponds to a model with 8 number of topics. Currently showing the 30 most relevant words for topic 1.*

6 Discussion

6.1 Topics Interpretation

In this section we are going to look deeper into the topics from the selected model and we are going to give an interpretation from it. This interpretation intends to give a broad idea of the current state of the conflict and the consequences that this brings for the UK and Europe.

As we can observe from Figure 7, we can attempt to interpret the topics and derive ideas from them.

Topic 0 contains the words: government, uk, gas, company, refugee, energy, home, visa, family, germany. This topic is more aimed at refugees and their worries. Like their legal status once they flee their country, their families or homes, essential things for living such as gas or energy. It could be inferred that this topic remains very present in the dataset since the news articles would be informing not only about the war but also the consequences that this is having on innocent people and how it affects their everyday life. Moreover, it could be inferred as well that the UK and Germany are receiving and helping refugees from the war.

Topic 1 contains the words: putin, military, force, president, nato, minister, zeleinskiy, moscow, defence, invasion. This topic relies heavily on the leaders of Russia and Ukraine, militia and words related to it such as defence or invasion. This topic simply demonstrates the main actors and contributors in the ongoing conflict.

Topic 2 contains the words: macron, le, price, pen, food, france, grain, vote, french, port. This topic shows a strong relation with France and activities that have a great impact on the economy, such as food, agriculture, and imports or exports of goods. It could be said that the words on this topic indicate that this areas of the economy would be specially affected in France.

Topic 3 contains the words: force, kyiv, civilian, mariupol, soldier, attack, kill, military, leave, region. This topic relies mostly in Ukraine. It mentions two of their cities along with words that could be associated as cause and effect. For instance, an attack by the military in one of those cities would cause civilians to leave the area, leaving them without home and possibly causing a large number of refugees.

Topic 4 contains the words: tell, work, family, time, know, leave, like, man, news, group. These word are related to the day to day activities of an adult, such as work, family, time. This implies that in the dataset a great number of articles create the space to talk about how the normal life of people in both regions is being affected by the conflict.

Topic 5 contains the words: medvedchuk, medvedev, african, world, china, africa, trophy, road, schevchenko, time. This topic contains Russian and Ukrainian names, most likely from public figures in their countries. That could mean involve-

ment from them in the conflict. The topic also contains words related to other nations, such as China and Africa. It could be said that a good number of news articles mention how these nations, in spite of not being geographically close to the conflict as other countries, are being affected by it.

Topic 6 contains the words: putin, soviet, world, like, time, think, right, time, new, gorbachev. Names from political figures from Russia are mentioned. If we link these with the rest of the words: world, time, think, new; it could be inferred how the government or politicians are having a great influence on the decisions that are being made about the war and its impact to the world.

Topic 7 contains the words: eu, sanction, putin, government, uk, european, abramovich, minister, include, party. There are words that indicate a more general consequence for the whole continent of Europe. Politicis is included as well in words like Putin, government, minister, and party. On the other hand, it can also be inferred that Russia is being sanctioned or punished in some way by Europe as a mean to try to persuade them to stop the war.

7 Conclusion

Firstly, it is worth mentioning that the pre-processing step, although is not part of the training of the model, it is essential to get good results. As such, it should be given its deserved importance when implementing an LDA algorithm. During our experimentation we observed that our results varied on quality if we applied some steps of the preprocessing stage on a slightly different way. For instance, using a different list of standardised stop words, removing the most common words that were generating noise in our results; or filtering news that did not share any similarities with the main topics of the rest of the dataset. Some additional steps for future research we recommend to try are generating bigrams and trigrams from the tokenised dataset, change the approach for lemmatisation of the tokenised dataset, or increase the size of the dataset.

In relation to the training of our model, as we observed during our experimentation, increasing the number of topics also increases the amount of time the model requires to be trained. For future research, it would be helpful to rely on sufficient computational resources if a more extensive analysis needs to be done in order to be able to speed up the entire pipeline to obtain the best results.

After tuning our model and choosing the best one in terms of topic interpretability, it is possible to infer general information about the main themes contained within our dataset. In our experiment, we obtained 8 topics from which we interpreted the main ideas behind each of the topics in relation to the current state of the conflict between Russia and Ukraine but also the consequences that this situation could have for the UK and Europe in general.

References

Albalawi, R., Yeap, T. H. & Benyoucef, M. (2020), 'Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis', *Frontiers in Artificial Intelligence* **3**.
URL: *https://www.frontiersin.org/articles/10.3389/frai.2020.00042*

Asmussen, C. B. & Møller, C. (2019), 'Smart literature review: a practical topic modelling approach to exploratory literature review', *Journal of Big Data* **6**(1), 1–18. Number: 1 Publisher: SpringerOpen.
URL: *https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0255-7*

Bird, S. (n.d.), 'Natural Language Processing with Python', p. 504.

Blei, D., Carin, L. & Dunson, D. (2010), 'Probabilistic Topic Models', *IEEE Signal Processing Magazine* **27**(6), 55–65. Conference Name: IEEE Signal Processing Magazine.

Blei, D. M. (2012), 'Probabilistic topic models', *Communications of the ACM* **55**(4), 77–84.
URL: *https://dl.acm.org/doi/10.1145/2133806.2133826*

Blei, D. M. (n.d.), 'Latent Dirichlet Allocation', p. 30.

Blei, D. M. & Lafferty, J. D. (2007), 'A correlated topic model of Science', *The Annals of Applied Statistics* **1**(1), 17–35. Publisher: Institute of Mathematical Statistics.
URL: *https://projecteuclid.org/journals/annals-of-applied-statistics/volume-1/issue-1/A-correlated-topic-model-of-Science/10.1214/07-AOAS114.full*

Caldara, D., Conlisk, S., Iacoviello, M. & Penn, M. (2022), 'The Effect of the War in Ukraine on Global Activity and Inflation'.
URL: *https://www.federalreserve.gov/econres/notes/feds-notes/the-effect-of-the-war-in-ukraine-on-global-activity-and-inflation-20220527.html*

Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C. & Blei, D. M. (n.d.), 'Reading Tea Leaves: How Humans Interpret Topic Models', p. 10.

Ferner, C., Havas, C., Birnbacher, E., Wegenkittl, S. & Resch, B. (2020), 'Automated Seeded Latent Dirichlet Allocation for Social Media Based Event Detection and Mapping', *Information* **11**(8), 376. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.
URL: *https://www.mdpi.com/2078-2489/11/8/376*

Honnibal, M. & Montani, I. (2017), spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Jacobi, C., van Atteveldt, W. & Welbers, K. (2016), 'Quantitative analysis of large amounts of journalistic texts using topic modelling', *Digital Journalism* **4**(1), 89–106.
URL: *http://www.tandfonline.com/doi/full/10.1080/21670811.2015.1093271*

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y. & Zhao, L. (2019), 'Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey', *Multimedia Tools and Applications* **78**(11), 15169–15211.
URL: *http://link.springer.com/10.1007/s11042-018-6894-4*

Li, N., Chow, C.-Y. & Zhang, J.-D. (2019), Seeded-BTM: Enabling Biterm Topic Model with Seeds for Product Aspect Mining, *in* '2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)', pp. 2751–2758.

Matsa, K. E., Silver, L., Shearer, E. & Walker, M. (2018), 'Western Europeans Under 30 View News Media Less Positively, Rely More on Digital Platforms Than Older Adults'.
URL: *https://www.pewresearch.org/journalism/2018/10/30/western-europeans-under-30-view-news-media-less-positively-rely-more-on-digital-platforms-than-older-adults/*

McCallum, A. K. (2002), Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet.

Nikolenko, S. I., Koltcov, S. & Koltsova, O. (2017), 'Topic modelling for qualitative studies', *Journal of Information Science* **43**(1), 88–102. Publisher: SAGE Publications Ltd.
URL: *https://doi.org/10.1177/0165551515617393*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A. & Cournapeau, D. (n.d.), 'Scikit-learn: Machine Learning in Python', *MACHINE LEARNING IN PYTHON* p. 6.

Rajaraman, A. & Ullman, J. D. (2011), *Mining of Massive Datasets*, Cambridge University Press, USA.

Ramage, D., Rosen, E., Chuang, J., Manning, C. D. & McFarland, D. A. (2009), Topic modeling for the social sciences, *in* 'Workshop on Applications for Topic

Models, NIPS'.
URL: *http://vis.stanford.edu/papers/topic-modeling-social-sciences*

Ray, S. K., Ahmad, A. & Kumar, C. A. (2019), 'Review and Implementation of Topic Modeling in Hindi', *Applied Artificial Intelligence* **33**(11), 979–1007. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/08839514.2019.1661576.
URL: *https://doi.org/10.1080/08839514.2019.1661576*

Rehurek, R. & Sojka, P. (2010), Software framework for topic modelling with large corpora, *in* 'IN PROCEEDINGS OF THE LREC 2010 WORKSHOP ON NEW CHALLENGES FOR NLP FRAMEWORKS', pp. 45–50.

Sievert, C. & Shirley, K. (2014), LDAvis: A method for visualizing and interpreting topics, *in* 'Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces', Association for Computational Linguistics, Baltimore, Maryland, USA, pp. 63–70.
URL: *http://aclweb.org/anthology/W14-3110*

theguardian / open platform - documentation / overview (n.d.).
URL: *https://open-platform.theguardian.com/documentation/*

Towne, W. B., Rosé, C. P. & Herbsleb, J. D. (2016), 'Measuring Similarity Similarly: LDA and Human Perception', *ACM Transactions on Intelligent Systems and Technology* **8**(1), 7:1–7:28.
URL: *https://doi.org/10.1145/2890510*

Xu, Z. & Raschid, L. (2016), Probabilistic Financial Community Models with Latent Dirichlet Allocation for Financial Supply Chains, *in* 'Proceedings of the Second International Workshop on Data Science for Macro-Modeling', DSMM'16, Association for Computing Machinery, New York, NY, USA, pp. 1–6.
URL: *https://doi.org/10.1145/2951894.2951900*

Zhao, W., Chen, J. J., Perkins, R., Liu, Z., Ge, W., Ding, Y. & Zou, W. (2015), 'A heuristic approach to determine an appropriate number of topics in topic modeling', *BMC Bioinformatics* **16**(13), 1–10. Number: 13 Publisher: BioMed Central.
URL: *https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-16-S13-S8*

Řehůřek, R. & Sojka, P. (n.d.), 'Gensim—Statistical Semantics in Python', p. 1.

8 Appendix One: Accompanying Archive and Instructions

8.1 Directory Structure

The repository can be found under the branch feature/lda-model-pycharm: https://git-teaching.cs.bham.ac.uk/mod-msc-proj-2021/nfq160/-/tree/feature/lda-model-pycharm.

The code files are under the directory "dissertation". This report files are under the directory "report". Furthermore, there is a txt file named "link to forms" which contains the url to the forms that were used for the human judgement experimentation. The generated html files to visualise the topics using pyLDAvis for each of the models generated in each run mentioned in this work can be found under dissertation/run x (where x is the number of the run).