Analysing Political Events on Twitter: Topic Modelling and User Community Classification

Anjie Fang

Submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

School of Computing Science

College of Science and Engineering
University of Glasgow

March 2019

Abstract

Recently, political events, such as elections or referenda, have raised a lot of discussions on social media networks, in particular, Twitter. This brings new opportunities for social scientists to address social science tasks, such as understanding what communities said, identifying whether a community has an influence on another or analysing how these communities respond to political events online. However, identifying these communities and extracting what they said from social media data are challenging and non-trivial tasks.

In this thesis, we aim to make progress towards understanding 'who' (i.e. communities) said 'what' (i.e. discussed topics) and 'when' (i.e. time) during political events on Twitter. While identifying the 'who' can benefit from Twitter user community classification approaches, 'what' they said and 'when' can be effectively addressed on Twitter by extracting their discussed topics using topic modelling approaches that also account for the importance of time on Twitter. To evaluate the quality of these topics, it is necessary to investigate how coherent these topics are to humans. Accordingly, we propose a series of approaches in this thesis.

First, we investigate how to effectively evaluate the coherence of the topics generated using a topic modelling approach. The topic coherence metric evaluates the topical coherence by examining the semantic similarity among words in a topic. We argue that the semantic similarity of words in tweets can be effectively captured by using word embeddings trained using a Twitter background dataset. Through a user study, we demonstrate that our proposed word embedding-based topic coherence metric can assess the coherence of topics like humans. In addition, inspired by the *precision at k* information retrieval metric, we propose to evaluate the coherence of a topic model (containing many topics) by averaging the top-ranked topics within the topic model. Our proposed metrics can not only evaluate the coherence of topics and topic models, but also can help users to choose the most coherent topics.

Second, we aim to extract topics with a high coherence from Twitter data. Such topics can be easily interpreted by humans and they can assist to examine 'what' has been discussed

on Twitter and 'when'. Indeed, we argue that topics can be discussed in different time periods and therefore can be effectively identified and distinguished by considering their time periods. Hence, we propose an effective time-sensitive topic modelling approach by integrating the time dimension of tweets (i.e. 'when'). We show that the time dimension helps to generate topics with a high coherence. Hence, we argue that 'what' has been discussed and 'when' can be effectively addressed by our proposed time-sensitive topic modelling approach.

Next, to identify 'who' participated in the topic discussions, we propose approaches to identify the community affiliations of Twitter users, including automatic ground-truth generation approaches and a user community classification approach. To generate ground-truth data for training a user community classifier, we show that the mentioned hashtags and entities in the users' tweets can indicate which community a Twitter user belongs to. Hence, we argue that they can be used to generate the ground-truth data for classifying users into communities. On the other hand, we argue that different communities favour different topic discussions and their community affiliations can be identified by leveraging the discussed topics. Accordingly, we propose a Topic-Based Naive Bayes (TBNB) classification approach to classify Twitter users based on their words and discussed topics. We demonstrate that our TBNB classifier together with the ground-truth generation approaches can effectively identify the community affiliations of Twitter users.

Finally, to show the generalisation of our approaches, we apply our approaches to analyse 3.6 million tweets related to US Election 2016 on Twitter. We show that our TBNB approach can effectively identify the 'who', i.e. classify Twitter users into communities by using hashtags and the discussed topics. To investigate 'what' these communities have discussed, we apply our time-sensitive topic modelling approach to extract coherent topics. We finally analyse the community-related topics evaluated and selected using our proposed topic coherence metrics.

Overall, we contribute to provide effective approaches to assist social scientists towards analysing political events on Twitter. These approaches include topic coherence metrics, a time-sensitive topic modelling approach and approaches for classifying the community affiliations of Twitter users. Together they make progress to study and understand the connections and dynamics among communities on Twitter.

Table of Contents

Abstract **i**

Acknowledgements **xvii**

1 Introduction **1**

1.1 Introduction 1

1.2 Motivations 3

1.3 Thesis Statement 4

1.4 Contributions 4

1.5 Origins of Material 5

1.6 Thesis Outline 6

2 Background **8**

2.1 Introduction 8

2.2 Topic Modelling 9

2.2.1 Background of Topic Modelling 9

2.2.2 Sampling 11

2.2.3 Variational Bayesian Inference 12

2.2.4 Topic Representation and Interpretation 13

2.3 Evaluation Methodology for Topic Modelling 14

2.3.1 Evaluating Predictability 15

2.3.2 Evaluating Topic Coherence 15

2.4 Text Classification 16

2.4.1 Ground-Truth Data . . . 17
2.4.2 Pre-processing . . . 17
2.4.3 Feature Selection . . . 18
2.4.4 Classification Algorithms . . . 19
2.4.5 Evaluation . . . 19
2.5 Applications in Social Science . . . 20
2.5.1 Topic Modelling in Social Science . . . 20
2.5.1.1 Document Inference . . . 21
2.5.1.2 Topic Examination . . . 21
2.5.2 The Use of Text Classification in Social Science . . . 22
2.6 Conclusions . . . 23

3 Analysing Political Events **24**
3.1 Introduction . . . 24
3.2 Twitter Topic Modelling . . . 25
3.2.1 Challenges . . . 25
3.2.2 Existing Topic Modelling Approaches . . . 26
3.2.2.1 Single Topic Assignment . . . 26
3.2.2.2 Pooling Strategy . . . 27
3.2.2.3 Enhancement using External Features . . . 29
3.3 Coherence Metrics . . . 32
3.3.1 Statistical Analysis of Coherence . . . 32
3.3.2 Semantic Coherence . . . 34
3.4 Twitter User Community Classification . . . 37
3.4.1 Manual Labelling Approaches . . . 38
3.4.2 Automatic Labelling Approaches . . . 39
3.4.3 Existing User Community Classification Approaches . . . 40
3.4.3.1 Classification Tasks . . . 40
3.4.3.2 Features used in User Classification . . . 42

3.5 Overview of our Approaches 46
3.6 Conclusions 47

4 Measuring Topic Coherence **49**
4.1 Introduction 49
4.2 Baseline Topic Coherence Metrics 50
4.3 Twitter-specific Topic Coherence Metrics 51
4.3.1 Using an External Resource for Coherence Metrics 51
4.3.2 Word Embedding-based Metrics 52
4.4 Metrics Comparison Methodology 54
4.4.1 Generating Topics 54
4.4.2 User Study 55
4.4.3 Agreement between Metrics and Human Assessors 56
4.4.4 Ranking of Topic Modelling Approaches 57
4.5 Evaluation of the Coherence Metrics 58
4.5.1 Datasets 58
4.5.2 Experimental Setup 59
4.5.2.1 Generating Topics using the Topic Modelling Approaches 60
4.5.2.2 CrowdFlower Quality Control 60
4.5.2.3 Metrics Setup 60
4.5.3 Research Questions 62
4.5.4 User Study Results 62
4.5.4.1 User Study Statistics 63
4.5.4.2 Agreement between Metrics and Human Assessors . . . 63
4.5.4.3 Ranking Comparison of the Topic Modelling Approaches 65
4.5.4.4 Summary 67
4.6 Evaluating the Global Coherence of Topic Models 68
4.6.1 `Coherence at n` Metric 69
4.6.2 Experiments 69

4.6.2.1 Datasets . . . 69
4.6.3 Experimental Setup . . . 70
4.6.3.1 Generating Topics . . . 70
4.6.3.2 Coherence Metrics Setup . . . 70
4.6.3.3 Research Questions . . . 71
4.6.4 Analysis of the Top Ranked Topics . . . 71
4.6.5 User Study . . . 74
4.6.5.1 Generating Topic Pairs . . . 74
4.6.5.2 Crowdsourcing Results . . . 75
4.7 Conclusions . . . 76

5 Time-Sensitive Topic Modelling **77**
5.1 Introduction . . . 77
5.2 Integrating the Time Dimension of Tweets . . . 78
5.2.1 Topical Trends . . . 79
5.2.2 Our TVB Approach . . . 81
5.2.3 Implementation of Time-Sensitive Topic Modelling . . . 83
5.2.3.1 Maximising the Lower Bound of a Document . . . 84
5.2.3.2 Expectation Maximization Algorithm . . . 87
5.3 Comparison to Baselines . . . 89
5.3.1 Comparison to Topic Over Time (ToT) . . . 89
5.3.2 Comparison to Twitter LDA . . . 90
5.4 Experimental Setup . . . 90
5.4.1 Datasets . . . 91
5.4.1.1 Ground-Truth Dataset . . . 91
5.4.1.2 US Election 2016 Twitter Dataset . . . 92
5.4.2 Generating Topics . . . 93
5.4.3 Evaluation Metrics . . . 94
5.4.3.1 Metric 1: Coherence Metrics . . . 94

5.4.3.2 Metric 2: Mixing Degree Metric 95
5.4.3.3 Metric 3: Trend Estimation Error 95
5.4.4 Research Questions 96
5.5 Evaluation of TVB 96
5.5.1 Results and Analysis on the GT Dataset 96
5.5.1.1 Topical Coherence and Topical Mixing Degree on the GT Dataset 97
5.5.1.2 A User Study of Mixed Topics 99
5.5.1.3 Topical Trends Estimation Error on the GT Dataset . . . 101
5.5.2 Results and Analysis on the USE Dataset 102
5.5.3 Summary of Results 104
5.5.4 Efficiency of the five Topic Modelling Approaches 105
5.6 Conclusions 106

6 Twitter User Community Classification 107

6.1 Introduction 107
6.2 Automatic Ground-Truth Generation Approaches 108
6.2.1 The Hashtag Labelling Approach 108
6.2.1.1 Hashtag Labelling Approach for IndyRef 109
6.2.1.2 Verification of the Hashtag Labelling using the Followee Network 110
6.2.2 The DBpedia Labelling Approach 111
6.2.2.1 Definitions of Communities 112
6.2.2.2 The Implementation of the DBpedia Labelling Approach 113
6.2.2.3 Two Baseline Labelling Approaches 115
6.2.2.4 The Three Generated Training Datasets 116
6.2.2.5 User Study 118
6.2.3 Summary 121
6.3 Topic-Based Naive Bayes — TBNB 122
6.3.1 Topics Analysis in IndyRef 122

6.3.2 Implementation of TBNB 122
6.4 Evaluation 124
6.4.1 Datasets 125
6.4.2 Experimental Setup 126
6.4.2.1 Classification Setup 126
6.4.2.2 Topic Setup in TBNB 127
6.4.2.3 Feature Selection 127
6.4.2.4 Metrics 128
6.4.3 Research Questions 129
6.4.4 Analysis of the IndyRef Community Classification Task 129
6.4.5 Analysis of the DBpedia Community Classification Task 132
6.5 Conclusions 137

7 Application on US Election 2016 139

7.1 Introduction 139
7.2 Data Collection 141
7.2.1 Labelled Data — Applying the Hashtag Labelling Approach 141
7.2.2 Unlabelled Data 142
7.3 Evaluating TBNB on US Election 2016 144
7.3.1 Experimental Setup 144
7.3.2 User Study for Twitter users' Candidate Preferences 145
7.3.3 Results of User Classification Experiments 147
7.3.4 Results of User Study 147
7.4 Applying TBNB on US Election 2016 149
7.5 Applying TVB on US Election 2016 151
7.5.1 Experimental Setup 151
7.5.2 Coherence Results 152
7.6 Analysing Topics in US Election 2016 153
7.6.1 Analysis of proClinton Topics 155

7.6.2 Analysis of proTrump Topics 157
7.6.3 Analysis of Topics across both Communities 159
7.6.4 Comparison to the Classical LDA Approach 161
7.7 Conclusions 163

8 Conclusions and Future Work **164**
8.1 Conclusions and Contributions 164
8.1.1 Contributions 165
8.1.2 Conclusions 167
8.2 Directions for Future Work 170
8.3 Closing Remarks 172

A Tables **174**

B Figures **178**

List of Tables

2.1 Examples of topics generated by LDA. 14

2.2 Symbols used in the definitions of the classification metrics. 20

3.1 The related work about different classification tasks on Twitter. 42

3.2 The related classification work using different features on Twitter. 43

4.1 The baseline topic coherence metrics introduced in Chapter 3. 51

4.2 Our proposed Twitter topic coherence metrics. 53

4.3 The *comparison units* of the three topic modelling approaches. 54

4.4 The details of the two used Twitter datasets for the study of the coherence metrics. 59

4.5 The number of the *comparison units* on the two used Twitter dataset. 59

4.6 The details of all the used topic coherence metrics. We use k to denote a thousand, i.e. $117k$ means $117,000$. 61

4.7 The results of the automatic topic coherence metrics on the **NYJ** dataset and the corresponding ranking orders. The values in the column of a metric are the coherence scores calculated by this metric. "$\times$" means no statistically significant differences (p $\leq$0.05, Wilcoxon signed-rank test, see Section 4.4.4) among the three topic modelling approaches. Two topic modelling approaches have the same rank if there are no significant differences between them. A metric is in bold if the ranking order of this metric matches/partly matches the order from the human assessors. 66

4.8 The results of the automatic topic coherence metrics on the **TVD** dataset and the corresponding ranking orders. The values in the column of a metric are the coherence scores calculated by this metric. "$\times$" means no statistically significant differences (p $\leq$0.05, Wilcoxon signed-rank test, see Section 4.4.4) among the three topic modelling approaches. Two topic modelling approaches have the same rank if there are no significant differences between them. A metric is in bold if the ranking order of this metric matches/partly matches the order from the human assessors. 67

4.9 Two used Twitter datasets for examining the top-ranked topics. 70

4.10 The comparison of coherence scores given by our coherence@n metric and human assessors. $*$/$(**)$ denote $p < 0.01$/$(p < 0.05)$ according to the Wilcoxon signed-rank test, compared to the smaller K. 75

5.1 The symbols used in our time-sensitive topic modelling approach. 83

5.2 Two used Twitter datasets for evaluating the topic modelling approaches. . . 91

5.3 The topic coherence, mixing degree and topic trends estimation error of the topic modelling approaches on the **GT** dataset. The subscripts indicate whether a given approach is significantly (p<0.05, using t-test) better than the other ones. The bold font indicates the highest value for each column. . 97

5.4 The results of our user study on mixed topics. 99

5.5 Topic samples from a TLDA model on the GT dataset, where the underlined words have a different topic theme from the rest of words in a topic. Note that we present a human assessor with the top 10 words of a topic in our user study. In this table, we only list the top 5 words for each topic. 100

5.6 Topic samples from a TVB ($\delta = 0.8$) model on the GT dataset, where the underlined words have a different topic theme from the rest of words in a topic. Note that we present a human assessor with the top 10 words of a topic in our user study. In this table, we only list the top 5 words for each topic. 100

5.7 The topical coherence and mixing degree of the 5 topic modelling approaches on the **USE** dataset with $\boldsymbol{K}$ **= 50**. The subscripts indicate whether a given approach is significantly (p<0.05, using t-test) better than the other ones. The highest score in each column is in bold. 103

5.8 The topical coherence and mixing degree of the 5 topic modelling approaches on the **USE** dataset with K **= 100**. The subscripts indicate whether a given approach is significantly ($p<0.05$, using t-test) better than the other ones. The highest score in each column is in bold. 103

5.9 The time consumption of the 5 topic modelling approaches on the USE dataset. 105

6.1 Agreement between the hashtag labelling approach and our followee network verification method. 111

6.2 Examples of combinations of DBpedia `predicates` & `objects` and the extracted DBpedia entities. 113

6.3 The Twitter public lists used in the baseline labelling approaches for generating ground-truth data. 115

6.4 The number of the labelled users of four communities generated using our DBpedia labelling approach. 117

6.5 The number of users in the training dataset for the DBpedia community classification task. 118

6.6 The comparison between human judgements and the DBpedia labelling approach. 120

6.7 Confusion matrix of the users' community labels between the DBpedia labelling approach and human assessors. 120

6.8 Two datasets for our user community classification tasks. (a) The dataset generated using our hashtag labelling approach. (b) The dataset generated using our DBpedia labelling approach and two baseline labelling approaches. 126

6.9 Topics and associated words in IndyRef. For each topic, the top 5 words (ranked by the conditional probabilities of words in topics) are listed in column "Associated Words". 127

6.10 The comparisons of 5 classifiers in terms of Precision, Recall and F1 in the IndyRef community classification task. The 5 classifiers are indexed by symbols ★, ■, ♣, ♠ and ▽. If a classifier significantly (p <0.05 in McNemar's test) outperforms another, we add the index symbols as subscripts on the "Accuracy" value of this classifier. The bold values indicate the best performance per column. 132

6.11 The community classification results using the three training datasets in the DBpedia community classification task. The rows with the grey background in these three tables indicate the best-performing classifier for a given training dataset. The bold values in (b) indicate the improved performance compared to the best performance in (a) while the bold values in (c) indicate the improved performance compared to the best performance in (b). The superscripts *, † or ⋆ indicate whether the best-performing classifier in (a), (b) or (c) is significantly (p <0.05) outperformed by the others with the superscript. 133

6.12 (a). The classification result of TBNB in the DBpedia community classification task. The superscript *, † or ⋆ indicates that whether TBNB significantly (p <0.05) outperforms NB_{BD}, SVM_{RBD} and SVM_{DBD} (listed in (b)), trained using the three training datasets, respectively. 135

6.13 Topics and associated words in the DBpedia training dataset. For each topic, the top 10 ranked words are listed in column "Associated Words". 137

7.1 The used hashtags for labelling Twitter users in US Election 2016. 142

7.2 The labelled and unlabelled datasets for the user community classification of US Election 2016. 142

7.3 The use of the labelled and unlabelled datasets in our application. 143

7.4 The Twitter user community classification results on US Election 2016. These results are obtained using our labelled dataset, where a 10-fold cross-validation is applied for all classifiers. We highlight in bold the highest values for reference. 148

7.5 The agreement between classifiers and human assessors on US Election 2016. 148

7.6 The number of users/tweets of the proClinton and proTrump communities in the unlabelled dataset. 149

A.1 The symbols used in Chapters 2, 3 and 4. 174

A.2 The combinations of `Predicate` & `Object` for the `ACA` and `MDA` communities. 175

A.3 The combinations of `Predicate` & `Object` for the `BE` and `PLT` communities. 176

A.4 Additional user community classification results for Chapter 6. 177

List of Figures

2.1 The plate notation of LDA. 10

2.2 The plate notation of LDA implemented by Variational Bayesian inference. 12

2.3 The five processes of text classification. 17

3.1 The approaches proposed in this thesis. 46

4.1 The designed user interface on CrowdFlower for obtaining the topic coherence judgements. 55

4.2 The associated tweets for the two shown topics in our user study. 56

4.3 Agreement between metrics and human assessors on topical coherence. . . 56

4.4 The confidence distribution of topic preferences from human judgements. (a) The NYJ dataset. (b) The TVD dataset. 63

4.5 The topic preference agreement between the human judgements and the 18 metrics. Each bar in the figure represents the agreement of a metric compared to the human judgements. 64

4.6 The Cohen's $kappa$ agreement between the human judgements and the 18 metrics. Each bar in the figure represents a Cohen's $kappa$ score of a metric compared to the human judgements. 64

4.7 The coherence of three types of topic models with different values of K over two datasets. 72

4.8 The coherence value distributions in the study of examining the top-ranked topics. 73

4.9 The CrowdFlower user interface for studying the top-ranked topics. 75

5.1 An example of topical trend. 80

5.2 Examples of two topics with two different trends. 80
5.3 Examples of two topics in a timeline. 81
5.4 The plate notation of (a) the classical LDA and (b) our TVB approach. . . . 82
5.5 The update directions in the EM implementation in (a) the classical VB approach and (b) our TVB approach. 88
5.6 The real and estimated topical trends estimated by the two topic modelling approaches on the GT dataset, where the x-axis and the y-axis represent the timeline and the density probability, respectively. 101

6.1 The user interface for obtaining the human judgements of community labels. 119
6.2 Example of a Twitter user. 121
6.3 The results of NB and TBNB in the IndyRef community classification task. (a), (b), (c) and (d) show the accuracy of TBNB where K is set to 5, 10, 20 and 30, respectively; (e) The accuracy of NB. In (a), the blue line overlaps with the black and purple lines. In (b) and (c), the blue lines overlap with the black line while the green one overlaps with the purple one. In (d), all the lines tend to overlap with each other except the red line. For example, TBNB_FR means that a TBNB classifier with FR feature selection approach. 130
6.4 Comparison of F_{test} and R_{test} for both the NB and TNBN classifiers ($K = 10$) in our IndyRef community classification task. In both (a) and (b), the blue lines overlap with the black line while the green one overlaps with the purple one. 131
6.5 Example of a Twitter user. 136

7.1 Our application on US Election 2016. 140
7.2 Our labelled and unlabelled datasets for US Election 2016. 143
7.3 The user interface of our Crowdflower user study for US Election 2016. . . 146
7.4 The number of tweets from the proClinton and proTrump communities over time in US Election 2016. 150
7.5 The number of tweets mentioning two candidates in the (a) proClinton and (b) proTrump communities in US Election 2016. 150
7.6 The coherence of topic models with different K. (a) topic models generated from the proClinton-related tweets; (b) topic models generated from proTrump-related tweets. 152

7.7 Topics extracted from proClinton (Topics 1-6) in US Election 2016 (generated by TVB). 154

7.8 Topics extracted from proClinton (Topics 7-12) in US Election 2016 (generated by TVB). 155

7.9 Topics extracted from proClinton (Topics 13-18) in US Election 2016 (generated by TVB). 156

7.10 Topics extracted from proTrump (Topics 1-6) in US Election 2016 (generated by TVB). 157

7.11 Topics extracted from proTrump (Topics 7-12) in US Election 2016 (generated by TVB). 158

7.12 Topics extracted from proTrump (Topics 13-18) in US Election 2016 (generated by TVB). 159

7.13 Topics extracted from the proClinton community using the classical LDA approach. 161

7.14 Topics extracted from the proTrump community using the classical LDA approach. 162

B.1 The 6 most coherent topics extracted from the ACA community in US Election 2016. 178

B.2 The 6 most coherent topics extracted from the MDA community in US Election 2016. 179

B.3 The 6 most coherent topics extracted from the BE community in US Election 2016. 180

B.4 The 6 most coherent topics extracted from the PLT community in US Election 2016. 180

B.5 Topics (7-12) extracted from proClinton using the classical LDA approach. 181

B.6 Topics (13-18) extracted from proClinton using the classical LDA approach. 181

B.7 Topics (7-12) extracted from proTrump using the classical LDA approach. . 182

B.8 Topics (13-18) extracted from proTrump using the classical LDA approach. 182

Acknowledgements

My PhD study at The University of Glasgow has become the most wonderful journey in my life, filled with excitement and joyfulness. This thesis is the end of this journey in obtaining my PhD. I would not finish it without the immense support I received in the past four years. I would like to take this opportunity to thank people who make this thesis possible and an unforgettable experience for me.

First and foremost, I would like to express my sincere gratitude to my PhD supervisors, Iadh Ounis, Craig Macdonald and Philip Habel, for their patience, enthusiasm, and immense knowledge. They have provided me with guidance, support and encouragement not only in my PhD research but also in my PhD life. Without their help, this work would have not been possible.

My sincere thanks also go to my friends and colleagues at the Terrier team and the school of computing science for collaboration and support: Xiao Yang, Graham MacDonald, Xi Wang, Ting Su, Haitao Yu, Xiaoyu Xiong, Jarana Manotumruksa, Richard McCreadie, Stuart Mackie, David Maxwell, Fatma Elsafoury and Colin Wilkie. I have learned a lot from them. It has been an honour and a pleasure to work with them.

I am grateful to Dell Zhang and Jeff Dalton for their thoughtful feedback and suggestions during my PhD viva, to Nick Craswell and Tat-Seng Chua for mentoring me in SIGIR 2017 Doctoral Consortium.

I am also thankful to Oleg Rokhlenko, Simone Filice and Nut Limsopatham, for mentoring me during my internship in their team at Amazon and leading me working on an exciting project.

Last but not least, I would like to thank my family, especially my mother, Qingyun Ping, for their endless support, belief and encouragement throughout my life.

Chapter 1

Introduction

1.1 Introduction

For decades, social scientists have sought to understand the connections and influences among communities[1]. For example, previous work has allocated individuals into communities in terms of vote preferences or ideological positions (e.g. left or right wing) in order to analyse how they respond during a political event, such as an election or a referendum (e.g. Mehrabian, 1998; Hillygus and Jackman, 2003; Vaccari et al., 2013; Barberá et al., 2015). Meanwhile, there have been work that investigated whether the media community played an important role in influencing policy-makers and citizens (Habel, 2012) or whether policy elites such as the business community exerted influences in an election (Hillman et al., 2004). On the other hand, social scientists also examined the content these communities communicated. For example, Bara et al. (2007) extracted topics[2] of conversations from the politicians to understand their policy positions using parliament debates. Jacobi et al. (2016) leveraged news articles to understand the trends and topics of the media community. Indeed, social scientists are interested in 'who' (i.e. communities) and 'what' they said (i.e. topics) to conduct communication studies.

Recently, the use of social media networks, such as Twitter[3], has increased dramatically and social media networks have emerged as the main channel for the mass public to express opinions, raise topic discussions or share ideas, especially during a political event,

[1] A community in this thesis means a group of people sharing the same profession (e.g. politicians) or having the same orientation in a political event (e.g. an election).

[2] A topic is a particular subject that people discuss. In this thesis, a topic is a word distribution extracted by using topic modelling approaches, e.g. LDA (Blei et al., 2003). It can be represented by the top-ranked words by its word distribution.

[3] `https://www.twitter.com`

such as an election or a referendum. Indeed, Twitter played an important role in the Scottish Independence Referendum 2014 (Feltwell et al., 2015), the UK General Election 2015 (Burnap et al., 2016), the US Election 2016 (Enli, 2017) and the UK European Union Membership Referendum (known as Brexit) (Llewellyn and Cram, 2016). The popularity of social media networks provides an opportunity to study 'who' said 'what' and 'when' they said it during a political event.

In this thesis, we aim to propose various approaches to assist the study of 'who' said 'what' and 'when' for a political event on social media networks. We propose effective approaches to identify the communities (i.e. 'who') and to extract their discussed topics (i.e. 'what') on social media networks during a political event[4] while taking into account the importance of the time dimension on social media networks (i.e. 'when'). To identify communities, we propose an effective user community classification approach for social media data[5]. In order to help social scientists to apply community classifiers, we also propose automatic ground-truth generation approaches to train these classifiers without human intervention. Since social media data is different from normal text corpora, such as news articles and books, the topics extracted from social media data can be incoherent, i.e. these topics are difficult for humans to understand and to interpret (Chang et al., 2009; Hong and Davison, 2010). Therefore, we propose a topic modelling approach to improve the coherence[6] of topics from social media data. Our proposed topic modelling approach is sensitive to the time trends of topics and thus generates topics that are easier for humans to interpret. Finally, to automatically assess the coherence quality of topics, we propose novel coherence metrics for evaluating topics generated from social media data. Indeed, the proposed coherence metrics can assist social scientists to select the most coherent topics from a large set of generated topics.

To show the usefulness of our proposed topic modelling approach, user community classification approaches and coherence metrics, we use them to analyse US Election 2016 from Twitter. We show how to use the proposed ground-truth generation and user community classification approaches when classifying Twitter users into communities in favour of two different presidential candidates. To examine the conversations and connections between the two communities, we apply our proposed topic modelling approach on tweets posted by users from these two communities. We show the usefulness of the proposed coherence metrics when evaluating and selecting topics in terms of coherence. Using these generated topics within communities, we explore the similarities and divergences among users in the two communities and analyse the behaviours of these communities during the election.

[4] A political event in this thesis typically means an election or a referendum.

[5] Posts that are created on social media networks

[6] The coherence of a topic is used to indicate how likely this topic can be interpreted by humans.

In the remainder of this chapter, we first discuss the motivations of this thesis in Section 1.2. In Section 1.3, we present the thesis statement followed by the contributions of the thesis in Section 1.4. The origins of material are listed in Section 1.5. Finally, we provide the thesis outline in Section 1.6.

1.2 Motivations

As mentioned in Section 1.1, the increased use of social media networks during a political event (e.g. in Burnap et al., 2016; Enli, 2017) is bringing new opportunities for social scientists to study the dynamics of communities, understand what these communities said and identify whether a community has an influence on another. Intuitively, while user community classification approaches can identify the community affiliations of users, i.e. identifying the 'who', a topic modelling approach can be applied to extract the topics (i.e. what) these communities discussed over the timeline of the event (i.e. 'when').

However, applying topic modelling and user community classification approaches on social media data are non-trivial tasks (e.g. in Derczynski et al., 2013; Cohen and Ruths, 2013) mainly because social media posts are short[7] and contain misspelt words and various peculiarities, which are different from documents in a normal text corpus. There are three main limitations: **1)** The topics generated from social media data using a topic modelling approach lack coherence (Chang et al., 2009; Hong and Davison, 2010), which can cause difficulties for humans to interpret. **2)** To evaluate the coherence of topics, the existing coherence metrics cannot work effectively for topics from social media data. **3)** When there are more and more political events happening on social media networks, there is a need to train many classifiers requiring effective approaches to automatically generate ground-truth data and classify users into communities for these events. In this thesis, we aim to overcome these limitations.

Overall, we propose effective approaches to model topics and classify communities from social media posts, which can assist social scientists to examine 'who' and 'what' they said in a political event. We also make use of the importance of the time dimension (i.e. 'when') on Twitter to generate topics with a higher coherence. In this thesis, we are motivated to answer the following questions: 1) how to evaluate the coherence of topics; 2) how to improve the coherence of topics; 3) how to generate reasonable ground-truth data and how to effectively classify social media users into communities.

[7]For example, a tweet at most contains 140 characteristics.

1.3 Thesis Statement

This thesis argues that the understanding of who said what and when within a political event on social media networks can be addressed through a series of approaches. In particular, identifying the 'who' will benefit from an automatic user community classification approach, while the 'what' and 'when' can be addressed through modelling the topics of conversations using a topic modelling approach that are inherently time-sensitive, and coherent in their nature.

In particular, by using word embedding, we can more accurately compute the semantic similarities of words thereby allowing to evaluate the coherence of topics automatically. By integrating the time dimension of social media posts (i.e. 'when'), a topic modelling approach can improve the coherence of the generated topics and hence the interpretability of these topics; Furthermore, by using contextual features, such as the mentioned entities, hashtags and discussed topics on social media networks, we can automatically obtain effective ground-truth data to develop effective user community classifiers. Together, these approaches can effectively identify 'who' (i.e. communities) discussed 'what' (i.e. topics) during a political event while taking into account the time dimension of an event ('when'). They assist users to analyse the connections and dynamics among communities.

1.4 Contributions

We contribute a series of approaches to identify communities and extract topics in order to analyse a political event on social media networks. These proposed approaches also contribute to the fields of topic modelling and user community classification tailored to social media data. We split our proposed approaches into three parts: topic coherence metrics, topic modelling and user community classification.

Topic coherence metrics. We propose various topic coherence metrics for evaluating topics generated from tweets by using topic modelling approaches. We conduct a large-scale user study to obtain human coherence judgements in order to identify the best-aligned coherence metric with human judgements. Building on the proposed coherence metric, we also contribute a metric that effectively evaluates the global coherence of a topic model containing the entire set of topics.

Topic modelling. We propose a novel topic modelling approach to improve the coherence of topics from social media posts by integrating the time dimension of the posts. We evaluate our approach compared to various baselines in terms of topical coherence. This

proposed approach allows to generate topics with a higher coherence. It can be particularly useful when a user wants to extract human-interpretable topics from social media posts.

User community classification. We first propose automatic community ground-truth generation approaches. We show that these approaches can be reasonably effective for conducting user community classification tasks for a political event. Our proposed approaches contribute to generate ground-truth data without human annotators. These proposed approaches can be particularly useful when there are more and more political events emerging online and there is a need to deploy many user community classifiers for different events. We also propose a topic-based user community classification approach, which can effectively classify Twitter users into communities. The resulting classifier is easy for social scientists to deploy as it does not require massive feature engineering[8].

An application on US Election 2016. We demonstrate the use of our proposed approaches when analysing US Election 2016 on Twitter. We show that our approaches help to analyse the connections and dynamics among two communities supporting the two presidential candidates in this election.

1.5 Origins of Material

The material in this thesis is based on a number of conference publications:

- Chapter 3: We introduced our approaches towards analysing a political event in the doctoral consortium of SIGIR 2017 (Fang, 2017).
- Chapter 4: We examined several existing topic coherence metrics and proposed Twitter topic coherence metrics in ECIR 2016 (Fang et al., 2016b). We proposed topic coherence metrics based on word embedding in SIGIR 2016 (Fang et al., 2016c). In addition, the proposed metric for evaluating a topic model was published in SIGIR 2016 (Fang et al., 2016a).
- Chapter 5: The proposed time-sensitive topic modelling approach is based on work published in ECIR 2017 (Fang et al., 2017) and work published in CIKM 2018 (Fang et al., 2018b).
- Chapter 6: The proposed automatic ground-truth generation approach using hashtags and the topic-based user community classification approach were first introduced in SIGIR 2015 (Fang et al., 2015a) and presented at SFDIA 2015 (Fang et al., 2015b).

[8]The Twitter REST API is needed to obtain some features of users on Twitter. Since there are usage limits on the Twitter REST API, it can be time-consuming to obtain such features.

- Chapter 7: The application of our proposed approaches for analysing US election 2016 on Twitter is published in Sage Open (Fang et al., 2018a).

1.6 Thesis Outline

The remainder of this thesis is organised as follows:

- Chapter 2 first introduces the background of topic modelling including two implementation approaches and several evaluation methods of topic modelling. We then explain the background of text classification. We also review the applications of topic modelling and classification in social science.
- Chapter 3 reviews the related work in terms of adapting topic modelling and user community classification on Twitter. We then introduce our proposed approaches towards analysing political events.
- Chapter 4 investigates topic coherence metrics for topics generated from Twitter. We examine the performance of the existing topic coherence metrics on Twitter data and propose several topic coherence metrics, such as the metric based on word embedding. On the other hand, we also propose a metric that evaluates the global coherence of a topic model.
- In Chapter 5, we propose a time-sensitive topic modelling approach for modelling topics from tweets, which can generate topics with a higher coherence. We explain the implementation of our time-sensitive topic modelling approach and compare our approach to the other state-of-the-art topic modelling approaches using two real-world Twitter datasets.
- Chapter 6 investigates the classification of users into communities. We first describe two ground-truth generating approaches: a DBpedia-based labelling approach and a hashtag-based labelling approach. In order to more effectively classify Twitter users into communities during a political event, we introduce a topic-based user community classification approach, which identifies word usage in both topics and communities.
- Chapter 7 describes an analysis of US Election 2016 on Twitter. We apply the proposed automatic ground-truth generation and user community classification approaches to associate Twitter users into communities supporting the two presidential candidates. We then apply our proposed topic modelling approach to extract topics from the two

communities. The proposed coherence metrics are first applied to evaluate the topic model and then to select the most coherent topics.

- Chapter 8 closes this thesis by highlighting the contributions and the conclusions of each chapter. We also discuss some possible future directions for our research.

Chapter 2

Background

2.1 Introduction

This thesis investigates approaches of topic modelling and user community classification for social media data. We use these approaches to address 'who' (i.e. communities) said 'what' (i.e. the discussed topics) and 'when'. In this chapter, we first introduce some background about topic modelling, including the used implementations and evaluation methods for topic modelling. For user community classification, we cover text classification since we consider our user community classification task as a text classification task. Moreover, for both topic modelling and user community classification, we introduce their applications in social science. The remainder of this chapter is organised as follows:

- Section 2.2 describes the background of topic modelling approaches and their two widely used implementations: the sampling-based and the Variational Bayesian-based approaches.
- Section 2.3 reviews how existing work evaluates the quality of the generated topics and topic models. In particular, we discuss the automatic evaluation of topics in terms of topical coherence.
- Section 2.4 describes how to conduct the text classification task in terms of generating ground-truth data, pre-processing data, applying feature selection and classification algorithms as well as the corresponding evaluation methods.
- Section 2.5 introduces the current use of topic modelling and text classification in social science. We provide a comprehensive survey about topic modelling and user community classification approaches in Chapter 3.
- Section 2.6 concludes this chapter.

2.2 Topic Modelling

In this section, we describe the background of topic modelling approaches in Section 2.2.1. We explain two widely used implementation approaches of topic modelling: the sampling-based and the Variational Bayesian-based approaches, in Sections 2.2.2 and 2.2.3, respectively. On the other hand, we also introduce how to present topics from a generated topic model and how to interpret the topics in Section 2.2.4.

2.2.1 Background of Topic Modelling

Topic modelling is a statistical approach, which can be used to extract topics occurring in a corpus of documents. Deerwester et al. (1990) first introduced a Latent Semantic Indexing (LSI) topic modelling approach, where Singular Value Decomposition (SVD)[1] is used to detect semantic topics from documents. Later, Hofmann (1999) proposed a probabilistic Latent Semantic Analysis (pLSA) model by deploying a latent class model[2], which represents the relations between topics, documents and words. Based on pLSA, Blei et al. (2003) proposed a Bayesian version of pLSA, called Latent Dirichlet Allocation (LDA), where Dirichlet prior distributions are deployed for both documents and topics. Among these three topic modelling approaches, LSI is often applied when comparing the similarity of documents and words in a low-dimensional space, for example, in the tasks of document classification (Baker and McCallum, 1998; Liu et al., 2004) or query-document matching in information retrieval (Hofmann, 1999; Wei and Croft, 2006; Manning et al., 2008b). On the other hand, pLSA and LDA can more intuitively identify the latent topics from a corpus, where a topic is modelled as a distribution over words (e.g. in Newman and Block, 2006; Mei et al., 2007; Zhao et al., 2011b). LDA is probably the most commonly used topic modelling approach in the literature. LDA can be easily adapted to many other variants in different applications, such as the Pachinko allocation (Li and McCallum, 2006), the dynamic topic models (Blei and Lafferty, 2006) and the online inference LDA (Canini et al., 2009). Hence, we use the LDA topic modelling approach for tweets as a baseline due to its widespread use. At the same time, we propose our tailored topic model variants built on LDA for social media data (See Chapter 5).

LDA is a generative probabilistic modelling approach. In an LDA topic model, there are K[3] topics. A topic k is represented by a multinomial distribution β_k (called the topic

[1] Non-negative Matrix Factorisation can also be used in LSI (Lee and Seung, 2001).

[2] In this latent class model, words in documents are variables while topics are latent variables.

[3] K is the number of topics in a topic model. K is a required parameter when applying a topic modelling approach. We list all the used symbols in this chapter and their descriptions in Table A.1.

word distribution, thereafter) over N words drawn from a Dirichlet prior η, where k is the index of the total number (K) of topics and N is the size of the word vocabulary. Let $\boldsymbol{W}$ denote all the words used in a corpus of D documents:

$$\boldsymbol{W} = \{\vec{w_1}, ..., \vec{w_d}, ..., \vec{w_D}\} \tag{2.1}$$

A document $\vec{w_d}$ can be represented using the following equation:

$$\vec{w_d} = \{w_{d,1}, ..., w_{d,i}, ..., w_{d,N_d}\} \tag{2.2}$$

where d is the d-th document of the corpus, N_d is the total number of words in the d-th document and $w_{d,i}$ is the i-th word identity in the d-th document. A document has a topic belief distribution (called the document topic distribution, thereafter) θ_d drawn from the Dirichlet prior α (denoted by $\theta_d \sim$ Dirichlet(α)). Accordingly, documents can then be generated using the generative processes of LDA:

1. Draw $\theta_d \sim$ Dirichlet(α), where $d \in \{1, ..., D\}$
2. Draw $\beta_k \sim$ Dirichlet(η), where $k \in \{1, ..., K\}$
3. For each word position d, i, where $d \in \{1, ..., D\}$ and $i \in \{1, ..., N_d\}$:
   - (a) Draw a topic assignment $z_{d,i} \sim \theta_d$
   - (b) Draw a word $w_{d,i} \sim \beta_{z_{d,i}}$

where $z_{d,i}$ is the assigned topic index for word $w_{d,i}$.

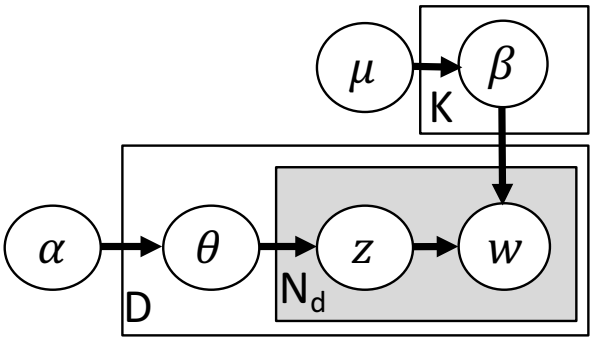

Figure 2.1: The plate notation of LDA.

The dependency of topics, documents and terms can be indicated by the plate notation as shown in Figure 2.1, where topics, documents and terms are represented as three plates. This plate notation indicates the relations among all the variables in LDA. The rectangle

groups the repeating variables (plates) and the number at the bottom of the rectangle is the number of repeating times. For example, for each document, the process of drawing a word is repeated N_d times. In practice, the topic modelling approaches are implemented so as to infer the two multinomial distributions ($\boldsymbol{\theta}$ and $\boldsymbol{\beta}$), given a corpus. In the following sections, we describe two main implementations of the LDA topic modelling approach: the sampling (see Section 2.2.2) and the Variational Bayesian-based (see Section 2.2.3) LDA implementation approaches.

2.2.2 Sampling

The sampling implementation approach follows the generative process of LDA and is based on the Markov Chain Monte Carlo (MCMC) (Gilks et al., 1995) method. In a typical sampling approach, such as the collapsed Gibbs sampling, each word of a document is assigned a topic index according to Equation (2.3):

$$p(z_{d,i} = k|\boldsymbol{z}_{-(d,i)}, \boldsymbol{w}) = \frac{n^{w_{d,i}}_{-(d,i),k} + \eta}{n_{-(d,i),k} + N\eta} \times (n^{d}_{-(d,i),k} + \alpha) \tag{2.3}$$

where $n^{w_{d,i}}_{-(d,i),k}$ is the frequency of word $w_{d,i}$ occurring in topic k and $n^{d}_{-(d,i),k}$ is the number of words from document $\vec{w_d}$ occurring in topic k not including the current one. This allows to construct a Markov Chain over the latent topics. After a number of iterations, $\boldsymbol{\beta}$ ($\{\beta_1, .., \beta_K\}$) and $\boldsymbol{\theta}$ ($\{\theta_1, .., \theta_D\}$) can be estimated from the converged Markov Chain using Equations (2.4) and (2.5):

$$\beta_{k,i} = \frac{n^{i}_{k} + \eta}{n_k + N\eta} \tag{2.4}$$

where n^{i}_{k} is the number of word w_i assigned to topic k and n_k is the total number of words assigned to topic k.

$$\theta_{d,k} = \frac{n^{d}_{k} + \alpha}{N_d + K\alpha} \tag{2.5}$$

where n^{d}_{k} is the number of words in the d-th document assigned to topic k and N_d is the total number of words in the d-th document. Meanwhile, the trained topic model from $\boldsymbol{w}$ can be used to estimate the topic distributions $\boldsymbol{\theta'}$ of a new corpus $\boldsymbol{w'}$ by using the following equations:

$$p(z_{d,i} = k|\boldsymbol{z'}_{-(d',i)}, \boldsymbol{w'}; \boldsymbol{z}, \boldsymbol{w}) = \frac{n^{i}_{k} + n^{w_{d',i}}_{-(d',i),k} + \eta}{n_k + n_{-(d',i),k} + N\eta} \times (n^{d'}_{-(d',i),k} + \alpha) \tag{2.6}$$

$$\theta_{d',k} = \frac{n^{d'}_{k} + \alpha}{N_{d'} + K\alpha} \tag{2.7}$$

where d' is the topic index of a document in $\boldsymbol{w'}$. Due to the simplicity of the sampling approach, many LDA variants have been proposed to handle different data, for example, author topic modelling (Rosen-Zvi et al., 2004), dynamic topic modelling (Blei and Lafferty, 2006), and Twitter-specific topic modelling (Zhao et al., 2011b). Due to the increasing amount of data, another type of implementation, the variational Bayesian approach, has been reported to be more efficient compared to the sampling approach (Hoffman et al., 2010). We introduce the Variational Bayesian-based implementation approach in the next section.

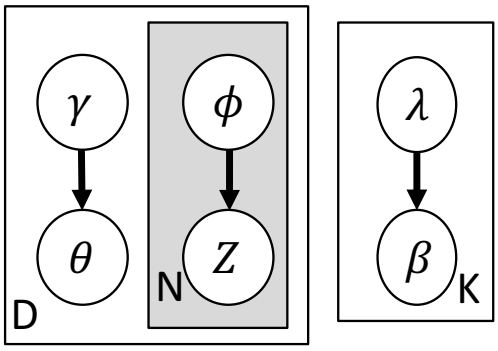

Figure 2.2: The plate notation of LDA implemented by Variational Bayesian inference.

2.2.3 Variational Bayesian Inference

While the sampling approach estimates the real posterior distributions (the topic word distributions $\boldsymbol{\theta}$ and the document topic distributions $\boldsymbol{\beta}$), a real posterior distribution is approximated by a variational distribution, i.e. minimising the distance between the real posterior distribution and its variational distribution, in the Variational Bayesian (VB) inference implementation approach (Blei and Jordan, 2004; Braun and McAuliffe, 2010). Specifically, an expectation maximization (EM) (Moon, 1996) algorithm is used to maximise the lower bound of the log-likelihood of all documents, which equivalently minimises the distances between the variational distributions and the true posterior distributions. As shown in Figure 2.2, the variational topic word distributions and document topic distributions are drawn by the variational Dirichlet priors λ and γ, respectively. The topic assignment is drawn by the words' topic belief $\phi_{D\times N\times K}$[4]. In the E step of EM, the variational Dirichlet prior γ_d of all documents are optimised together with the words' topic belief of documents ϕ_d by using Equations (2.8) and (2.9):

$$\phi_{d,i,k} \propto exp\{E_q[log\beta_{k,i}] + E_q[log\theta_{d,k}]\} \tag{2.8}$$

[4]Similar to Section 2.2.1, D, N and K are the number of documents, the size of vocabulary and the number of topics, respectively, indexed by d, n and k.

$$\gamma_{d,k} = \alpha + \sum_{i,k} \phi_{d,i,k} \tag{2.9}$$

where E_q indicates the variational expectation, which is calculated in the E step of EM. In the M step of EM, $\phi_{D\times N\times K}$ is then used to update the variational Dirichlet prior λ of $\boldsymbol{\beta}$ by using Equation (2.10):

$$\lambda_{k,i} = \eta + \sum_{d,i,k} \phi_{d,i,k} \tag{2.10}$$

Finally, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ can be obtained when the lower bound converges. The main advantage of the VB approach compared to the sampling approach is that the lower bound converges much more quickly than the sampling approach especially on large datasets (Blei and Jordan, 2004; Braun and McAuliffe, 2010). Moreover, the VB approach can be intuitively implemented in parallel since the updates of γ_d and ϕ_d among documents do not impact each other, while the sampling approach cannot be easily parallelised as it is intrinsically sequential (Asuncion et al., 2009). Due to the increasing volume of social media data and its dynamicity, it could be argued that the VB approach offers various advantages for those interested in analysing and interpreting discussions on social media as events transpire. Indeed, to deal with a large dataset, Hoffman et al. (2010) proposed an online version of LDA[5], which is implemented by using the VB approach. In this thesis, we compare the performance differences between these two topic modelling implementations (i.e. the sampling-based and variational Bayesian-based LDA approaches) and propose more effective topic modelling variants based on Variational Bayesian inference for Twitter data in Chapter 5.

2.2.4 Topic Representation and Interpretation

In a topic model, a topic is typically a multinomial distribution over words (i.e. a topic word distribution):

$$\beta_k = \{p(w_1|z=k), ..., p(w_i|z=k), p(w_n|z=k)\} \tag{2.11}$$

By ranking the conditional probabilities $p(w_i|z = k)$, we can obtain the top n words for each topic. Usually, the top-10 words are selected to represent the content of a topic. This representation has been used in many prior work, for example, in Blei et al. (2003); AlSumait et al. (2009); Chang et al. (2009); Newman et al. (2010); Zhao et al. (2011b). Hence, we use the same topic representation method to examine the content of topics from a topic model. Table 2.1 lists three examples of topics, which are generated from news articles, books and Twitter data, respectively. By examining and understanding the top 10 words,

[5]In this thesis, we do not address the online LDA approach as this approach is proposed to improve the efficiency, while we aim to improve the coherence of topics.

we can interpret the subject of these topics. For example, we can easily connect the words "music" and"film" to the subject of "art". However, the level of the interpretability of topics varies. For instance, the topic of "Theresa May" also involves words from the event of the Rio Olympic games 2016: "#iamteamgb" and "#rio". When social scientists extract a large number of topics, it can be time-consuming to examine all these topics. An effective automatic coherence metric is needed to help social scientists to quickly select and focus on the most coherent topics. We further introduce the method of evaluating the coherence of topics in the next section.

Table 2.1: Examples of topics generated by LDA.

Topic Subject	Top n words
Art (Blei et al., 2003)	new film show music movie play musical best actor first
Furniture (Newman et al., 2010)	furniture chair table cabinet wood leg mahogany piece oak louis
Theresa May[6]	#theresamaypm #iamteamgb #rio #brexit minister people speech @harryslaststand michelle britain

2.3 Evaluation Methodology for Topic Modelling

When a topic modelling approach is applied on a corpus, a trained topic model can be generated. The obtained topic model contains topic word distributions representing the latent topics and document topic distributions indicating the topic affiliations of documents. A generated topic model is often used for other tasks, such as document classification (Blei et al., 2003) or document matching in information retrieval (Yi and Allan, 2008), which means that the quality of a topic model can be indirectly evaluated by evaluating the performance of the resulting classifiers and retrieval systems. However, this does not evaluate the natural characteristics of a topic model. The direct use of topic modelling is to group documents under the same topic subject or to predict the topic's belonging of an unseen document using the generated topic model. Hence, a statistical method can be used to evaluate how well the generated topic model preforms in estimating the topics of unseen held-out documents. On the other hand, as mentioned in Section 2.2.4, topic modelling approaches enable to examine the content of topics from a corpus. To interpret the extracted topics, it is necessary to automatically evaluate the interpretation quality of these topics. In this section, we discuss two main evaluation methods of topic modelling: evaluating the predictability of topic models and assessing the coherence of topics.

[6] Sample of tweets posted between June and July in 2016 in the UK and collected using the Twitter Streaming API.

2.3.1 Evaluating Predictability

To evaluate the predictability of a topic model, we can estimate the probability of the held-out probability of the unseen documents (or the per-word likelihood) using a topic model that is generated using training documents. This held-out probability of documents is shown in Equation (2.12):

$$p(\boldsymbol{W}|\boldsymbol{W'}) = p(\boldsymbol{W}|\boldsymbol{\beta}, \alpha) \tag{2.12}$$

where $\boldsymbol{W}$ is the unseen held-out documents and $\boldsymbol{W'}$ is a set of training documents. $\boldsymbol{\beta}$ is the topic word distributions. The log probability of a held-out document probability $p(\boldsymbol{W}|\boldsymbol{W'})$ is then used to calculate the perplexity of a topic model and then to evaluate the quality of a topic model. Measuring perplexity has been previously widely used in many probabilistic models (e.g. in Gildea and Hofmann, 1999; Minka and Lafferty, 2002; Rosen-Zvi et al., 2004). Wallach et al. (2009) refined this predictability evaluation method and proposed a method to more effectively and efficiently estimate the log probability of held-out documents, which can be generally used in different types of topic models, such as in Mimno et al. (2009); Arora et al. (2013); Patterson and Teh (2013); Cong et al. (2017). At the same time, this automatic evaluation method can be applied to tune the parameter K, the number of topics (e.g. in Hinton and Salakhutdinov, 2009; Zubir et al., 2017; Karami et al., 2018). The best K can be selected when a topic model reaches the highest held-out probability. Although, it is popular to use the perplexity to measure the predictability of topic models, it cannot help to examine how good the generated topics are when users try to interpret the content of topics. This is because that evaluating the predictability of a topic model does not connect to the way humans interpret a topic (c.f. Section 2.2.4). Hence, in this thesis, we aim to propose evaluation metrics that help social scientists to assess the quality of topics in terms of coherence. We further discuss the evolution of topical coherence metrics in the following section.

2.3.2 Evaluating Topic Coherence

The topic word distribution in a topic model can help to interpret the content of topics extracted from a corpus using the topic representation introduced in Section 2.2.4. The topic modelling approaches are expected to generate coherent topics that are easy for humans to interpret. However, not all of the generated topics in a topic model are coherent (Steyvers and Griffiths, 2007; AlSumait et al., 2009). The most intuitive way of evaluating topical coherence is to involve human annotators and let humans judge the coherence level of a

latent topic[7]. This manual evaluation method has been widely used in the literature. For instance, Rosen-Zvi et al. (2004) and Steyvers et al. (2004) proposed a probabilistic author-topic model and evaluated the coherence of the generated topics from a human perspective. They reported that the topics in the author-topic model were representative of the topic content. Similar analyses were conducted in other topic modelling approaches, such as the labelled LDA approach (Ramage et al., 2009; Liu et al., 2009). When generating topic models from a corpus with a large size, the manual method of coherence evaluation is not suitable since there is a large number of topics that are generated and human annotations can be expensive to obtain. Hence, it is necessary to develop an automatic topic coherence metric that can quickly evaluate the coherence quality of topics in a similar manner to humans. Such a metric can help to select the most coherent topics among a set of generated topics and thus can save time when examining the content of topics. The topic coherence metric offers an automatic method to evaluate a topic modelling approach in terms of topical coherence. In this thesis, we aim to generate human-interpretable topics from social media data. We hence evaluate the coherence quality of topics by proposing effective topic coherence metrics for social media data. We review more existing topic coherence work in Chapter 3 and investigate topic coherence metrics for Twitter data in Chapter 4.

In this section, we have introduced some necessary background about topic modelling, how to represent the content of the generated topics and how to evaluate the generated topics. While the topic modelling approaches assist to interpret the content of topics extracted from a corpus, the text classification approaches can help to group documents into categories or to identify their labels. Next, we introduce some background on text classification and their use in social science applications.

2.4 Text Classification

In this thesis, we study the user community classification on social media networks. A user community classifier can identify the communities (i.e. the 'who') from social media posts. Since the profile and posts of a user on social media networks are text-based, we introduce supervised text classification in this chapter. In machine learning, classification is the problem of associating a data instance into one label of a set of categories (also called classes). A data instance can be a user or an object. For example, in sentiment classification, a document can be grouped into positive or negative categories (e.g. in Pang et al., 2002). Users can be classified in terms of their ethnicities or political orientations (Pennacchiotti

[7]The topic presented to humans is represented by the top n words from its word distribution, c.f. Section 2.2.4.

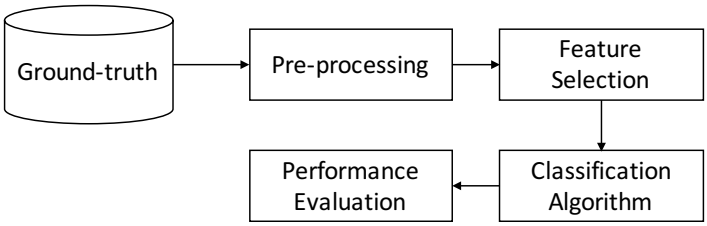

Figure 2.3: The five processes of text classification.

and Popescu, 2011). In text classification, a data instance is usually represented by text. A classifier is then trained by learning rules and patterns from text. In the following, we explain how to train a text classifier in terms of 5 processes shown in Figure 2.3.

2.4.1 Ground-Truth Data

The first step of a text classification task is to collect a set of data instances with known ground-truth labels, also known as the training dataset. The ground-truth labels indicate in which classes instances belong to, i.e. every instance is associated with a ground-truth label. The ground truth labels of data instances are crucial in supervised text classification since they are required in the training process so as to learn rules and patterns.

2.4.2 Pre-processing

The training data is usually pre-processed so that data instances can be transferred into vectors before applying a classification algorithm. There are several commonly used pre-processing procedures:

- **Tokenisation**: Tokenisation is the process of splitting a given document or a sentence into pieces, called tokens. A token is usually a word. Sometimes, it can also be a phrase or a hashtag on Twitter. Usually, the punctuations are also removed after tokenisation.
- **Removing Stopwords**: A stopword is a commonly used word, such as “the”, “a”, “an”, “in”, etc. Stopwords occur across classes in training data. They are usually not informative when training a classifier. Therefore, stopwords are usually removed.
- **Stemming**: Stemming is the process of reducing the inflectional forms of a word to their word stem. For example, words “cat”, “cats”, “cats’” and “cat’s” can be converted to the word stem “cat”. Since these different forms of the word “cat” represent the same meaning, they are usually reduced to their word stem in the pre-processing step of text classification. In the literature, the commonly used stemmers are the:

Lovins stemmer (Lovins, 1968), Porter stemmer (Porter, 1980) and Paice/Husk stemmer (Chris et al., 1990). We apply the Porter stemmer in this thesis due to its high effectiveness (Manning et al., 2008a).

2.4.3 Feature Selection

A feature is an attribute shared by all the data instances. In text classification, a word is a feature. Before applying a classification approach, a data instance is transferred to a feature vector, where an instance is represented by a vector of words. Commonly, the collection of a training dataset is represented by a matrix, where a row in the matrix is a data instance and a column indicates a feature. A cell in the matrix can be a boolean, the term frequency (TF) or the term frequency-inverse document frequency (i.e. `TF-IDF`) of a word feature. However, the number of features can be huge, which can result in a matrix with high dimensions. A higher dimension of training data can have a higher computational cost. To reduce the number of dimensions, feature selection approaches can be applied. The commonly used feature selection approaches select features by (1) Term Frequency ($Frequency$), (2) Log Probability Ratio ($LogProbRatio$), (3) Exponential Probability Ratio ($ExpProbRatio$), (4) Odds Ratio ($OddsRatio$) or (5) Weighted Odds Ratio ($WeightedOddsRatio$) (for futher details, see Mladenic and Grobelnik, 1999) as shown in the following equations:

$$Frequency(w) = TF(w) \tag{2.13}$$

$$LogProbRatio(w) = log\frac{p(w_{pos})}{p(w_{neg})} \tag{2.14}$$

$$ExpProbRatio(w) = e^{p(w_{pos})-p(w_{neg})} \tag{2.15}$$

$$OddsRatio(w) = log\frac{p(w_{pos}) \times (1 - p(w_{neg}))}{(1 - p(w_{pos})) \times p(w_{neg})} \tag{2.16}$$

$$WeightedOddsRatio(w) = p(w) \times OddsRatio(w) \tag{2.17}$$

where w is a word and $p(w)$ is its probability in a corpus. $p(w_{pos})$ and $p(w_{neg})$ are word probabilities in the positive and negative classes, respectively. Only the top-ranked features by these feature selection approaches are usually selected for text classification. For multi-class classification, the `one-vs-rest` strategy (e.g. used in Weston et al., 1999) can be applied where a single classifier is trained for each class to identify whether an instance belongs to a class or to the rest of classes. The trained classifier can still be seen as a binary classifier and therefore these feature selection approaches can still be applied in multi-class classification using the aforementioned `one-vs-rest` strategy. On the other hand, the

one-vs-one strategy can also be applied for multi-class classification, where a classifier is trained between each of two classes and the total number of the classifiers is $n(n-1)/2$ (n is the total number of the classes). In this thesis, we apply the one-vs-rest strategy since it can be computationally expensive to apply the one-vs-one strategy.

2.4.4 Classification Algorithms

A series of machine learning approaches can be used in text classification, e.g. Naive Bayes, Decision Tree, Support Vector Machine, or Neural Network (see Aggarwal and Zhai, 2012, for a comprehensive survey). Naive Bayes is a statistical model based on the known prior probabilities and the conditional probabilities in classes. Naive Bayes is easy to implement and deploy. Naive Bayes has been reported to have a good performance in various text classification tasks (e.g. in Lewis, 1998; McCallum et al., 1998). In particular, a multinomial Naive Bayes model is as competitive as Support Vector Machine (Rennie et al., 2003). Support Vector Machine learn a hyperplane (decision surface) that separates the classes. The Support Vector Machine classifier has been reported to perform effectively in text classification (Joachims, 1998). The Decision Tree classifier identifies the classes of the instances by learning simple decision rules from the training dataset. A Decision Tree model is easy to interpret. However, a Decision Tree model can be sensitive to small fluctuations in the training data and hence can be easy to overfit. On the other hand, a Neural Network classifier is a network model, where the units of the input layer are the word features of the data instances and the output units are the classes. There are hidden layers between the input and output layers in Neural Network and the units between two adjacent layers are fully connected. A commonly used Neural Network model is Multi-layer Perceptron (Kruse et al., 2013). Although, it has been shown that Multi-layer Perceptron performs effectively in text classification (Ruiz and Srinivasan, 1998), it has also some disadvantages. For instance, Multi-layer Perceptron requires to tune a number of hyperparameters (e.g. the number of layers) and the computational cost can be high. We refer the interested reader to the survey by Aggarwal and Zhai (2012) for further details about these classifiers. In this thesis, we apply these commonly used classification approaches as baselines in order to evaluate our user community classification approaches (see Chapter 6).

2.4.5 Evaluation

Various classification evaluation metrics have been used in classification tasks. We mainly introduce the commonly used metrics: Precision, Recall, F-measure, and Accuracy. The F-measure is the harmonic mean of precision and recall. We use the F1 metric, where precision

and recall are evenly weighted. The equations of how these metrics measure the performance of a classifier are shown as follows:

$$Precision = \frac{TP}{TP + FP} \tag{2.18}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.19}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.20}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.21}$$

where TP, FP, FN and TN are described in Table 2.2.

Table 2.2: Symbols used in the definitions of the classification metrics.

Symbols	Description
TP	Ture Positive: The number of instances correctly assigned into this class.
FP	False Positive: The number of instances incorrectly assigned into this class.
FN	False Negative: The number of instances incorrectly rejected into this class.
TN	True Negative: The number of instances correctly rejected into this class.

2.5 Applications in Social Science

Social scientists have embraced topic modelling and text classification in many applications ranging from studying the political positions (Barberá, 2015) of individuals to identify whether a political speech supports a given legislation (Thomas et al., 2006). In the following, we review examples of such applications of topic modelling and test classification in Sections 2.5.1 and 2.5.2, respectively.

2.5.1 Topic Modelling in Social Science

While topic modelling is widely used in computing science, it also plays an important role in social science. Its applications can be summarised in two categories: 1) Document inference. The topic affiliation of an unseen document can be computed and predicted using a generated topic model. Thus, documents can be categorised into topics. 2) Topic examination. Topic modelling can be applied on a corpus and then the extracted topics can help to examine the discussed topics in a corpus. Next, we discuss the use of topic modelling in social science in terms of document inference and topic examination.

2.5.1.1 Document Inference

In topic modelling, inference is the process of estimating topic word distributions and document topic distributions. In social science, topic modelling variants have been proposed to solve specific problems. For example, Roberts et al. (2014) developed a structural topic modelling approach by taking covariates (e.g. age and gender) into account. This model allows to further examine the relations between covariates and topics, which were used in political studies to analyse topics from social media posts (Lucas et al., 2015). Similar to the theory of topic modelling, Barberá (2015) proposed a Bayesian model which inferred the ideology positions (i.e. left and right wing) as latent variables. Instead of using words as observations, Barberá (2015) inferred the ideology positions of individuals using the users' followers on Twitter. The proposed model was first applied to group individuals in terms of their ideology positions in the US Election 2012 and then to analyse which community (i.e. the left or right-wing community) was more likely to participate in the topic discussions online during the election (Barberá et al., 2015). On the other hand, McCallum et al. (2005) proposed an author-recipient topic modelling approach in order to discover the relations between users[8] in a social network by using the key attributes in emails. In a similar study, topic modelling was applied to examine the behaviours of authors when citing other work (Ding, 2011). Indeed, topic modelling is useful in social science since it can be flexibly applied to extract latent variables. In this thesis, we do not investigate the inference of social media posts. Instead, we aim to apply topic modelling on social media data to extract and study the topics discussed by Twitter users during a political event.

2.5.1.2 Topic Examination

The most common use of topic modelling in social science is to examine topics extracted from documents using the topic representation method described in Section 2.2.4. This allows social scientists to quickly obverse and summarise the content of topics generated from a corpus (e.g. in Klebanov et al., 2008). Prior work has examined topics in news articles or newspapers for information discovery (e.g. in Blei et al., 2003; Steyvers et al., 2004). Ramabhadran et al. (2007); Quinn et al. (2010) extracted topics from parliament speeches to identify whether a politician is interested in a particular topic. Similarly, Jacobi et al. (2016) found that topic modelling was a useful tool to extract and then study the dynamics of topics from a large corpus of news articles.

[8]A user can be represented by the concatenation of documents they have written, e.g. emails or Twitter posts. A user can be seen as a document in topic modelling.

In recent studies, topic modelling was reported to be promising when dealing with social media data. For instance, Lucas et al. (2015) studied how to apply topic modelling for tweets in different languages to examine the content of topics. Sokolova et al. (2016) identified election-related events from Twitter using topic modelling and showed that these generated topics can be effectively used to conduct further election-related analysis. Further, Ryoo and Bendle (2017) studied the social media strategies of the campaigns of the two presidential candidates in US Election 2016 by examining the discussed topics on Twitter. Indeed, social media networks can provide up-to-date and trending data (Quinn et al., 2010). For analysing the data, topic modelling can be applied to automatically generate topics from social media posts, which can be then used to examine and summarise the content of topics (Boyd-Graber et al., 2017). Although it has been shown that the use of topic modelling on social media is promising in social science, the topics (e.g. the 'Theresa May' topic in Table 2.1) from a topic model can be difficult for humans to interpret (Newman et al., 2010; AlSumait et al., 2009), especially when generating topics from Twitter data (Chang et al., 2009; Zhao et al., 2011b). In order to help social scientists to examine topics from Twitter, it is necessary to develop a topic modelling approach that can generate topics with a higher coherence from social media data. In this thesis, we aim to propose an effective tailored topic modelling approach that generates coherent topics from social media data (see Chapter 5).

2.5.2 The Use of Text Classification in Social Science

Text classification has been widely applied in social science. For instance, Agrawal et al. (2003) developed a classifier to identify whether a newspaper supports or opposes a given topic. Kwon et al. (2007) classified the attitudes expressed in public comments (e.g. supporting or opposing) towards government documents (e.g. regulations). Moreover, Thomas et al. (2006) investigated a classification approach, which can automatically identify whether a speech supports or opposes a proposed legislation. Due to the popularity of social media networks, in particular Twitter, the classification of the Twitter users' characteristics and attributes is becoming a rapidly developing research topic. Indeed, there were efforts aimed at classifying users' ages, ethnicities and genders using features such as their last names, their following networks, the words in their posted tweets, or a combination thereof (e.g. in Al Zamal et al., 2012; Rao et al., 2010; Mislove et al., 2011). For example, Barberá (2016) generated a ground-truth data by matching the Twitter usernames to the voter registration files in the US. The generated ground-truth data is used to train a classifier for classifying partisanship and other demographic variables. Chen et al. (2015) took advantage of the friends and follower networks, user profiles, and user images to derive politically relevant characteristics about the Twitter users. Moreover, Pennacchiotti and Popescu (2011) showed

the performance of classifying the political orientations of the Twitter users can be enhanced by leveraging the Twitter users' profile descriptions. Indeed, most users indicate their occupations, interests and organisational affiliations in their profiles or tweets. Nowadays, more and more people are joining the discussions of political events on social media networks (e.g. in Burnap et al., 2016; Enli, 2017). Hence, a popular resulting task covered in the literature encapsulates the identification of the community affiliations of Twitter users (e.g. in Al Zamal et al., 2012; Cohen and Ruths, 2013). As we will show later in this thesis, we investigate effective approaches for training and developing effective classifiers categorising users into communities during a political event on social media networks (see Chapter 6).

2.6 Conclusions

We have introduced the necessary background for topic modelling and text classification. We first introduced two implementation approaches of topic modelling, namely the sampling and variational Bayesian-based approaches in Section 2.2.2 and Section 2.2.3, respectively. We explained the reason why we choose to build our topic modelling approaches (further introduced in Chapter 5) upon the Variational Bayesian-based approach. Second, we discussed methods to evaluate the topic modelling approaches including the evaluation of the topic models' predictability and the evaluation of the topical coherence. We also described text classification and the common steps needed to tackle a text classification task. Finally, we reviewed the applications of topic modelling and text classification in social science. We showed that topic modelling and text classification are widely used tools in social science.

In the next chapter, we review the related work of topic modelling and user community classification for addressing social media data while positioning our thesis in the literature.

Chapter 3

Analysing Political Events

3.1 Introduction

In the previous chapter, we discussed the necessary background about topic modelling and text classification. We also reviewed applications of topic modelling and text classification in social science. We focus our work on Twitter data, since we aim to monitor political events as discussed by both the politicians and the public. In this chapter, we start by reviewing several approaches from the literature, which improve the coherence of topics on either normal corpora or Twitter data. We also survey the details of several existing metrics that allow to evaluate the coherence of the generated topics. Next, we review existing approaches for automatically generating ground-truth data for developing user community classifiers. We introduce several existing approaches used to automatically classify Twitter users into communities. Building on the current limitations of the existing work, we provide an overview of our proposed approaches towards analysing political events on Twitter. The detailed outline of the chapter is as follows:

- Section 3.2 reviews the existing topic modelling approaches for improving the coherence of topics on Twitter data. We summarise the related work into three categories: approaches using single topic assignment for each tweet, approaches using pooling strategies across tweets and approaches using other features.
- Section 3.3 reports the related work about measuring the coherence of topics generated using topic modelling approaches. This includes metrics based on statistical analysis of coherence and metrics based on semantic similarity.
- Section 3.4 first reviews the related work about generating ground-truth data for user

classification including manual and automatic labelling methods. Second, we review the existing work about classifying Twitter users into communities.

- Section 3.5 describes an overview of our proposed approaches towards analysing political events on Twitter including a tailored time-sensitive Twitter topic modelling approach, new metrics tailored to the evaluation of the coherence of Twitter topics, and new approaches to generate large training datasets for developing user community classifiers for political events.

- Section 3.6 provides concluding remarks for this chapter.

3.2 Twitter Topic Modelling

Recently, Twitter has become the main channel for the mass public to express preferences and opinions, to raise topic discussions and to obtain the latest news, especially during social events, such as referenda or elections. Yet despite the ubiquity of social media, scholars still wrestle with the appropriate tools for best capturing the topics of discussion conveyed over these platforms (Zhao et al., 2011b; Mehrotra et al., 2013). To this end, the topic modelling approaches have been deployed on tweet corpora to examine topics and summarise discussions from tweets. However, due to the differences between tweets and normal text documents (e.g. news articles and books), it is challenging to model topics from tweets. In this section, we discuss these challenges and review existing topic modelling approaches for Twitter data.

3.2.1 Challenges

Tweet corpora are different from the traditionally used text corpora, such as news articles and books. News articles usually have rich text information. On the other hand, a tweet on Twitter is restricted to 140 characters[1]. Commonly, a tweet only expresses a single topic or contains snippets of a conversation. This can cause problems when applying topic modelling, as mentioned in Hong and Davison (2010); Zhao et al. (2011b); Yan et al. (2013). In a topic model, a document can be seen as a mixture of multiple topics (i.e. a document has a distribution over topics, θ, see Section 2.2.1), which implies that a document can contain multiple topics. This assumption can fit with news articles as there can be multiple topics discussed in a news article. However, topic modelling might not be intuitively applied on

[1]The limit was doubled to 280 in November 2017.

tweets directly since a tweet mostly has one topic. In addition, tweets can contain hashtags (e.g. #indyref, the event of the Scottish Independence Referendum 2014), URLs, the mentions of Twitter users (e.g. @theresa_may, Twitter handle of Mrs Theresa May), misspelt words and abbreviations. These peculiarities of tweets can raise difficulties when interpreting the content of topics. Next, we review the related work in improving the coherence of topics generated from tweets.

3.2.2 Existing Topic Modelling Approaches

Many approaches have been proposed to improve the coherence of topics generated from tweets. We discuss the existing work using three categories: 1) single topic assignment, 2) pooling strategy and 3) topic model enhancement using external features. Note that not all of the reviewed approaches are not initially tailored to Twitter. We also discuss their suitability for Twitter data.

3.2.2.1 Single Topic Assignment

In the generative process of topic modelling, each word in a document is assigned a topic index according to its document topic distribution. Since a tweet (which can be seen as a document) is unlikely to contain multiple topics, a single topic index can be assigned to all the words in a single tweet. This method is called single topic assignment. Initially, this method was proposed for normal text corpora in Gruber et al. (2007). Gruber et al. assumed that words in the same sentence should have the same topic and successive sentences can also share the same topics. The proposed model was reported to generate topics with a better quality and to better predict unseen documents, compared to the classical LDA. This single topic assignment strategy can help to deal with tweets. Zhao et al. (2011b) first proposed a similar method in a Twitter-specific topic modelling approach (called Twitter LDA). In this topic modelling approach, a single user u (u is the user index out of the total number of users U)[2] is associated with several tweets. A user u can have a topic distribution θ_u over topics. In the generative process, each tweet is assigned a topic index (z) and the words ($w_{d,i}$) of this user are drawn by using the word distribution of the same topic (z) shown as follows[3]:

1. Draw $\theta_u \sim$ Dirichlet(α), where $u \in \{1, ..., U\}$
2. Draw $\beta_k \sim$ Dirichlet(η), where $k \in \{1, ..., K\}$

[2]We list all of the used symbols in this chapter and their descriptions in Table A.1.

[3]This generative process is simplified. In a Twitter-specific topic model, a background word distribution was deployed to distinguish the background words (i.e. words that occur in multiple topics) from the topic-specific words (Zhao et al., 2011b).

3. For each tweet d in user u ($u \in \{1, .., U\}$):
   (a) Draw a topic assignment $z_{u,d} \sim \theta_u$
   (b) For each word position d, i in tweet d:
      i. Draw a word $w_{d,i} \sim \beta_{z_{u,d}}$

Indeed, this topic modelling approach can effectively model tweet corpora and generate human interpretable topics, as evaluated by human judgements (Zhao et al., 2011b). Due to the advantage of the single topic assignment, it has been widely applied in many other applications. For example, Zhao et al. (2011a) proposed a probabilistic model to discover topical key-phrases assuming that a single tweet contains only one topic. Diao et al. (2012) applied the single topic assignment strategy to obtain user-level topic distributions, which was then used to detect bursty topics on Twitter. However, the single topic assignment strategy has two limitations. First, even though a tweet is likely to have only one topic, it may still contain words that are used by other topics. Therefore, simply assigning all words in a tweet to a single topic index could lead to topics that are mixed with the others and therefore incoherence. Second, this method might not work effectively if a Twitter user does not have many tweets. The topic distribution of a user can be difficult to infer if there are few tweets posted by each user. This is because there are too few words in a user to determine their topic distribution. In fact, to obtain an event-related dataset on Twitter, the Twitter Streaming API is used by setting keywords indicative of the event (e.g. IndyRef). In the collected dataset, the average number of tweets per user is rather low. In this case, the single topic assignment method might not be suitable for modelling topics of political events on Twitter. We use Twitter LDA (using the single assignment method) as a baseline for our proposed time-sensitive topic modelling approach in Chapter 5.

3.2.2.2 Pooling Strategy

Instead of the single topic assignment method, another way of overcoming the shortness of tweets is the pooling strategy. In such a strategy, a number of tweets are combined together as a virtual document by concatenating tweets. There are two advantages for the pooling strategy. First, the occurrences of words are increased in a combined virtual document and they are higher than those of words in a single tweet, which makes it easier for topic modelling. Second, the combined virtual documents are likely to have multiple topics, which align with the assumption of topic modelling, i.e. a document is a mixture of multiple topics. Initially, Rosen-Zvi et al. (2004) proposed an author-topic model for a corpus of conference papers. The strategy can be seen as document pooling, i.e. all the papers by the same author

are aggregated together as a virtual document for this author. When dealing with a tweet corpus, Hong and Davison (2010) also applied an author-wise pooling strategy and showed that it can provide a higher performance in a classification task compared to the topic modelling of a corpus of unpooled tweets. Indeed, this author-wise pooling strategy helps to improve the quality of generated topic in terms of coherence (Weng et al., 2010; Zhao et al., 2011b). On the other hand, Mehrotra et al. (2013) proposed several other pooling strategies. For example, tweets can be grouped together if they share the same bursty terms (burst-wise), if they are posted during a short period of time (e.g. one hour), or if they share the same hashtags[4] (hashtag-wise). Their experiments indicated that the hashtag-wise pooling strategies helped to generate a better topic model (as evaluated by clustering metrics) in comparison to the other pooling strategies. However, there are two limitations when applying these pooling strategies. First, it can be difficult to set the number of tweets to combine into each virtual document. Second, this strategy might not be suitable when a corpus does not have a lot of hashtags or when the average number of tweets per user is low. In Section 4.6, we evaluate the performance of the pooling strategy in generating coherent topics from Twitter data.

Apart from grouping tweets together in a pooling strategy, words can also be grouped together, e.g. bi-term (i.e. two-word combination). Yan et al. (2013) and Cheng et al. (2014) both proposed bi-term topic modelling approaches to deal with short tweets. These topic modelling approaches rely on the co-occurrence of bi-term in a whole corpus rather than the occurrence of words at the document level. Specifically, in the generative process, each bi-term (w_1 & w_2) is assigned a topic assignment z drawn from the topic distribution (θ) of the whole corpus, as shown below:

1. Draw a topic distribution $\theta \sim$ Dirichlet(α) for the whole corpus
2. Draw $\beta_k \sim$ Dirichlet(η), where $k \in \{1, ..., K\}$
3. For each biterm b in the corpus:
    (a) Draw a topic assignment $z_b \sim \theta$
    (b) For each word position w_1 & w_2 in bi-term b:
        i. Draw a word $w_1 \sim \beta_{z_b}$
        ii. Draw a word $w_2 \sim \beta_{z_b}$

The bi-term model was reported to overcome the shortness of tweets and to generate more coherent topics (Yan et al., 2013; Cheng et al., 2014; Xia et al., 2015). However, Quan et al. (2015) showed that the bi-term topic modelling does not necessarily generate meaningful topics for short texts. This is because generating bi-terms might bring less discriminative

[4]Many topics can be discussed in a hashtag event. Hence, a hashtag might not be used to identify a topic.

word co-occurrence knowledge (Xia et al., 2015; Li et al., 2016). On the other hand, Xia et al. (2015) proposed a discriminative bi-term topic modelling approach, where words in a corpus are treated differently at the topic and document levels. Although the bi-term topic model brings a higher performance in terms of topical coherence, it has limitations when modelling topics in a political event. In particular, the number of bi-terms in a corpus can be huge and thus the method might not be efficient for a corpus containing a large number of tweets. In addition, since we also aim to integrate the time dimension of tweets during the topic modelling, the resulting model might become too complex and cumbersome to use. Therefore, we do not consider the bi-term topic modelling approach in the remaining of this thesis. Next, we review several topic modelling enhancements, including one integrating the time dimension of documents.

3.2.2.3 Enhancement using External Features

In order to improve the quality of topic models, external features can be used during the modelling process. Such features can be tags of documents, the created timestamps of documents or word representations (i.e. word embedding). Next, we review the related work using external features to improve the coherence of topics from tweets.

Wang and McCallum (2006) developed a topic model supervised by the posted time of articles, called Topics Over Time (ToT). Documents posted near the same time are more likely to discuss the same topic. In such a model, one additional notion, time, was added to the standard LDA topic model. A topic k also has a distribution τ_k over time, which can be seen as the popularity of topics. The topic time distribution τ_k is actually a beta distribution parametrised by two shape parameters ρ_k^1, ρ_k^2. In the generative process, for each word position, both words and their generated timestamps are assigned, as follows:

1. Draw $\theta_d \sim$ Dirichlet(α), where $d \in \{1, ..., D\}$
2. Draw $\beta_k \sim$ Dirichlet(η), where $k \in \{1, ..., K\}$
3. Draw $\tau_k \sim \rho_k^1, \rho_k^2$, where $k \in \{1, ..., K\}$
3. For each word position d, i, where $d \in \{1, ..., D\}$ and $i \in \{1, ..., N_d\}$:
   (a) Draw a topic assignment $z_{d,i} \sim \theta_d$
   (b) Draw a word $w_{d,i} \sim \beta_{z_{d,i}}$
   (b) Draw a timestamp $t_{d,i} \sim \tau_{z_{d,i}}$

The ToT approach is implemented based on the Gibbs sampling approach (introduced in Section 2.2.2). Different from Equation (2.3) used in a standard sampling approach, ToT uses the following equation to assign a topic assignment to a word:

$$p(z_{d,i} = k|z_{-(d,i)}, \boldsymbol{w}) = \frac{n^{w_{d,i}}_{-(d,i),k} + \eta}{n_{-(d,i),k} + N\eta} \times (n^{d}_{-(d,i),j} + \alpha) \times \frac{(1 - t_{d,i})^{\rho^1_k - 1} \times (t_{d,i})^{\rho^2_k - 1}}{B(\rho^k_1, \rho^2_k)} \quad (3.1)$$

where the first two multiplicands are the same as the multiplicands from Equation (2.3). The third multiplicand in Equation (3.1) is added to ToT to integrate time. In fact, different topics happened in a different time period (e.g. in Lee et al., 2011; Lu and Yang, 2012). A word should not be assigned a topic index if this topic is not popular in a time period. This approach was reported to not only generate topics with higher quality but also to provide popularity of topics over time. However, ToT also has limitations when extracting topics from tweets. First, as additional observed variables, the timestamps of words within a document are set to the created timestamp of a document. This means that a document contains a number of different words associated with identical timestamps. Therefore, it can be argued that the topic model should depend more on the occurrences of words than those of the timestamps since the usage of words in different topics brings more useful information to distinguish these topics. As can been seen in Equation (3.1), the importance of words and timestamps in ToT is treated equally. In Chapter 5, we propose a new time-sensitive topic modelling approach that addresses the aforementioned limitations and compare it to ToT and other existing baselines from the literature.

Ramage et al. (2009) proposed a labelled LDA model for web pages where each page has a human-labelled tag. Such tags have influences on the topic document distribution using a Bernoulli distribution. This method could be applied for tweets, where a hashtag can be seen as a tag. However, in a labelled LDA model, the number of topics is set to be the same as the number of tags. This strategy might not work effectively during a political event where many hashtags are used. In particular, different hashtags should not be treated equally. For example, #ge2016 (a hashtag on Twitter) generally indicates the UK general election 2016 while #theresamay and #brexit have a relatively smaller scope compared to #ge2016. Additionally, the topic of #ge2016 encapsulates both #theresamay and #brexit. Hence, we do no use this approach in our thesis.

Traditionally, a topic model is built from bags-of-word documents. However, the bags-of-words model is usually very sparse when a document is very short. Sridhar (2015) proposed to use the neural network-trained word embeddings (i.e. a dense word representation) for modelling topics from short texts. In this model, a Gaussian mixture model containing K components (i.e. topics) is trained using the embeddings of words in the vocabulary regardless of the level of the document. A topic index is assigned to a document by comparing the distance between this document and the topics of the Gaussian mixture model. Such an

enhancement is reported to generate topics with higher coherence. Instead of using word embeddings as inputs of the topic model, Li et al. (2016) proposed a topic model, which considered the word semantic similarity during the sampling process. Specifically, when assigning a topic index to a word of a document in the sampling approach, the semantic similarity (calculated by the similarity of the word embeddings) between this word and all the other words in the vocabulary are involved. By doing so, semantically similar words tend to be in the same topic and the model can generate topics with higher coherence. A similar approach was proposed in Nguyen et al. (2015). On the other hand, Shi et al. (2017) showed that word embeddings were learned from a local context window while LDA models documents in a global view. Hence, topic models and word embeddings can be trained together and benefit each other. Shi et al. proposed a skip-gram topical word embedding model to learn word representations for each topic. These word representations then linked semantically similar words together in the topic modelling. In general, these approaches, which use word embeddings, have been reported to improve the topical coherence (e.g. Sridhar, 2015; Li et al., 2016; Shi et al., 2017).

We do not use word embeddings in our work. There are three main reasons for this. First, we aim to propose a topic modelling approach, which can be quickly applied to extract trending topic discussions during a political event. Topic modelling with word embeddings might not be easy to deploy by social scientists since word embedding training requires tweet samples and it can be time-consuming to collect and pre-process a large volume of tweets for different political events. For example, Sridhar (2015) used a 10% random sample of tweets (approximately 15 million tweets corresponding to only 2 weeks of data) to train word embeddings for their topic model. Second, topic modelling with word embeddings (e.g. in Li et al., 2016) tends to assign semantically similar words into the same topic. However, the resulting topic might not be a real topic although the topic is coherent. Third, we aim to integrate the time dimension of tweets in topic modelling. As mentioned earlier, words can be assigned to the same topic if they are used in the same time period. However, when using topic modelling with word embeddings, words can be assigned to the same topic when they are semantically similar. Hence, a topic modelling approach integrating time works differently from the one that uses word embeddings. When combined, the resulting topic model might be complex and not easy to interpret by social scientists.

Thus far, we have reviewed several existing topic modelling approaches. We showed that these approaches have limitations when extracting topics from a political event on Twitter. For example, the single assignment method could lead to incoherent topics when used with Twitter LDA. Moreover, although ToT does integrate time, it is not clear whether it will work effectively for tweets and whether the added time features have necessarily the

same importance as the words that are typically used in topic modelling. To better capture the dynamics of topics discussed over time in a political event, we propose a new tailored time-sensitive topic modelling approach in Chapter 5, which aims to extract coherent topics while suitably controlling the importance of time with respect to the words.

3.3 Coherence Metrics

The topic modelling approach can be used to address 'what' topics have been discussed during a political event on Twitter. To evaluate the quality of the generated topics, it is important to apply suitable metrics to automatically evaluate the coherence of the generated topics from tweets. In Section 2.3, we have introduced the evaluation methods of topic models including methods based on document predictability and topical coherence. Since we aim to generate human-interpretable topics from tweets, we focus on evaluating the topical coherence and review the existing work in evaluating the coherence. In this section, we review two types of coherence metrics: metrics based on statistical analysis and metrics based on semantic similarity.

3.3.1 Statistical Analysis of Coherence

A topic is a distribution over words in a topic model. Therefore, the quality of a topic can be evaluated by analysing its topic word distribution. Cao et al. (2009) applied a method to calculate the distance of pairwise topic distributions (e.g. topics k & k') using distance metrics, such as Kullback-Leibler (KL) divergence, shown as follows:

$$KL(\beta_k, \beta'_k) = \sum_i \beta_{k,i} log \frac{\beta_{k,i}}{\beta_{k',i}} \tag{3.2}$$

where KL is the Kullback-Leibler divergence. $\beta_{k,i}$ is the probability of word w_i (i indicates the word index in the vocabulary) in topic k. If the topic distributions are similar (i.e. a low distance or divergence), the topics are likely to be mixed[5] and to be incoherent. A higher average distance among topics indicates that the topics' distributions are different from each other, which suggests a good topic model. By using such distance/divergence measurements, Mei et al. (2007) proposed an automatic approach to generate coherent labels for interpreting topics. Later, Arun et al. (2010) argued that both topic word and document topic distributions should be considered to measure topic similarity. Both methods were reported to effectively

[5]Topics are similar since they share mutual words, which also indicate that topics are likely to be mixed.

evaluate the topic distribution and can also be used to select an appreciate setting for the number of topics (K). Further, AlSumait et al. (2009) measured the coherence of topics by measuring the distance between a topic and three defined incoherent/meaningless topics. If a topic is close to any of the three incoherent/meaningless topics, this suggests that this topic is less coherent. These three defined topics are the uniform distributions over words[6], a vacuous distribution over words[7] and a background distribution over documents[8]. In the following, we explain how coherence metrics work by using the three topic distributions:

Metric U. In the topic's term distribution, all terms have an equal and constant probability, which is unlikely to be meaningful nor to be easily interpreted by a human. A typical **u**niform term distribution β_{uni} is defined in Equation (3.3), where i is the word index, N is the total number of word vocabulary and $p(w_i)$ is the word (w_i) probability in a topic.

$$\beta_{uni} = \{p(w_1), p(w_2), ..., p(w_N)\},\ p(w_i) = \frac{1}{N} \tag{3.3}$$

Therefore, using metric U (**u**niform), the coherence of topic k is calculated as follows:

$$Coherence^{U}(k) = KL(\beta_{uni}||\beta_k) \tag{3.4}$$

Metric V. A real topic should contain a unique collection of highly used words distinguishing this topic from the other topics. A topic is less coherent if a topic is mixed. A **v**acuous term distribution θ_{vac} represents a mixed term distribution, in which the term probability reflects the frequency of the term in the whole corpus. β_{vac} is defined by Equation (3.5), where d is the document index and D is the total number of documents.

$$\beta_{vac} = \{p(w_1), p(w_2), ..., p(w_N)\},\ p(w_i) = \sum_{k=1}^{K} \beta_{i,k} \times \frac{\sum_{d=1}^{D} \theta_{d,k}}{D} \tag{3.5}$$

In metric V (**v**acuous), we compute the coherence as follows:

$$Coherence^{V}(k) = KL(\beta_{vac}||\beta_k) \tag{3.6}$$

Metric B. A real topic should represent documents within a semantically coherent theme. If a topic is close to most of the documents in the corpus, it is likely to be less meaningful and less coherent. Whereas the previous two distributions use terms to define the incoherent distribution of a topic, the topic distribution over documents can also reflect

[6] All word probabilities are same within the uniform distribution.
[7] The top-ranked words in the vacuous distribution are similar to the top-ranked words in the whole corpus.
[8] A flat topic distribution which responses all the documents in the corpus.

the quality of the topic (AlSumait et al., 2009). A topic's document distribution ϑ_k is defined in Equation (3.7) and a typical **b**ackground document distribution ϑ_{gb} is defined in Equation (3.8), where $\vec{w}_d$ means the d-th document.

$$\vartheta_k = \{p(z = k|\vec{w}_1), p(z = k|\vec{w}_2), ..., P(z = k|\vec{w}_D)\} \tag{3.7}$$

$$\vartheta_{gb} = \{p(\vec{w}_1), p(\vec{w}_2), ..., p(\vec{w}_D)\},\ p(\vec{w}_i) = \frac{1}{D} \tag{3.8}$$

Hence, using metric B (**b**ackground), the coherence of a topic is indicated as:

$$Coherence^B(k) = KL(\vartheta_{gb}||\vartheta_k) \tag{3.9}$$

Although AlSumait et al. (2009) showed that it was promising to use the three defined meaningless topic distribution to rank the generated topics, it is still not clear how this method aligns with human judgements, especially when evaluating the coherence of topics from tweets. Therefore, in Chapter 3, we use these mentioned metrics (i.e. metrics U, V and B) as baselines in our study of coherence metrics for Twitter data (See Section 4).

3.3.2 Semantic Coherence

A semantic coherence-based method is promising since it measures the coherence of topics by considering the semantic similarity of the top n words from a topic distribution. For instance, the word "Glasgow" is semantically similar to the word "Scotland" as Glasgow is a big city in Scotland. A topic is coherent if most of its top n ranked words are semantically similar. Newman et al. (2009, 2010) first proposed a topic coherence metric that can measure the topics' coherence. In this method, a topic k is represented by the top n words ($\{w_1, w_2, ..., w_n\}$). A word pair of a topic is composed of any two words from the topic's top n words. The coherence of a topic is then measured by averaging the semantic similarities of all word pairs, as follows:

$$Coherence(topic) = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j=i+1}^{n} f_{SS}(w_i, w_j) \tag{3.10}$$

where the coherence function f_{SS} calculates the semantic similarity using external resources, such as Wordnet[9] and Wikipedia pages[10]:

[9] https://wordnet.princeton.edu
[10] https://dumps.wikimedia.org

Using WordNet. WordNet is a lexical ontology in a hierarchical structure. It groups words into synsets ($117k$ in total), where synsets are linked according to their semantic and lexical relations (Fellbaum, 1998). There are a number of semantic similarity and relatedness methods in the literature, which can be used to calculate the semantic similarity between two words in WordNet. Among them, the method designed by Leacock and Chodorow (1998) (denoted as LCH[11]) and the one designed by Jiang and Conrath (1997) (denoted as JCN[11]) are especially useful for discovering lexical similarity (Newman et al., 2010). According to Newman et al. (2010), the semantic similarities of two words (w_i and w_j) using LCH and JCN are computed by the following equations:

$$LCH(w_i, w_j) = -\frac{sp(w_i, w_j)}{2 \times D} \tag{3.11}$$

where $sp(w_i, w_j)$ indicates the shortest path of words w_i and w_j in WordNet and D is the maximum depth of WordNet.

$$JCN(w_i, w_j) = \frac{1}{IC(w_i) + IC(w_j) - 2 \times IC(lcs(w_i, w_j))} \tag{3.12}$$

where IC is information context ($IC(w) = -log\ p(w)$) of a word in a corpus and $lcs(w_i, w_j)$ is the least common subsumer of words w_i and w_j in WordNet. For instance, the lcs of words "boat" and "car" is "vehicle". Apart from these two methods, Newman et al. (2010) also showed that the method from Lesk (1986) (denoted as LESK[11]) performs well in capturing the similarity of word pairs. LESK calculates the number of overlapping words in the definitions of words (e.g. w_i and w_j) in WordNet. Hence, in Chapter 4, we deploy topic coherence baseline metrics where the approaches of LCH, JCN and LESK (i.e. implementing f_{SS}) are used to compute the semantic similarities of words and then to identify the coherence scores of topics.

Using Wikipedia. Wikipedia pages have been previously used as background data to calculate the semantic similarities of words (Rus et al., 2013; Recchia and Jones, 2009). There are two widely used approaches in the existing literature to calculate the semantic similarities of words using Wikipedia pages: Pointwise Mutual Information (PMI) and Latent Semantic Analysis (Landauer et al., 1998) (LSA). PMI is an effective method to capture semantic similarity (Rus et al., 2013). Newman et al. (2009, 2010) reported that the performance of PMI was close to human judgements when assessing the coherence of topics generated from news articles. Here the PMI scores of word pairs from Wikipedia pages are computed using:

[11] Abbreviations that were used in the original papers.

$$f_{SS}(w_i, w_j) = PMI(w_i, w_j) = log\frac{p(w_i, w_j)}{p(w_i) \times p(w_j)} \tag{3.13}$$

where $p(w_i, w_j)$ is the joint probability of two words and $p(w_i)$ is the probability of a single word in a corpus. On the other hand, LSA is also commonly used to calculate the semantic similarities of words. For example, Landauer et al. (1998) applied LSA to study the relatedness between the segments. In particular, Chang et al. (2009); Newman et al. (2010) leveraged LSA to evaluate the coherence of topics (generated from news articles and newspapers) and to identify whether a topic is interpretable to humans. Specifically, a LSA model contains dense vectors of words (same as LSI, introduced in Section 2.2.1). In a LSA-based metric (e.g. Newman et al., 2010), the semantic similarities of words $f_{SS}(w_i, w_j)$ can be computed using distance metrics (e.g. cosine similarity) between the words' vectors as follows:

$$f_{SS}(w_i, w_j) = cosine(V_{w_i}, V_{w_j}) \tag{3.14}$$

where V_{w_i}, V_{w_j} are the vector representations of w_i, w_j in a pre-trained LSA model. To form the aforementioned PMI and LSA-based coherence metrics, the PMI scores and the LSA model are first trained using Wikipedia pages (i.e. background data).

Newman et al. (2009, 2010) evaluated both the PMI and LSA-based metrics and reported that the PMI-based metric best aligns with human judgements when assessing the coherence of topics generated from news articles. However, it is worth investigating these metrics for evaluating the coherence of topics generated from tweets. First, the used corpus in their work was news articles, which is different from a tweets corpus. Indeed, it is not clear how well these metrics work for topics generated from tweets. In particular, the Wikipedia pages may not be suitable as an external source for evaluating the coherence of topics generated from tweets as the usages of words in Wikipedia pages and tweets can be very different. For example, tweets can contain hashtags, mentions of Twitter users and abbreviations, which can be included in a generated topic, such as the following example topic[12]:

> #theresamaypm #brexit minister @DavidDavisMP speech economy @teamgb #rio watch thanks

The mentions (e.g. @DavidDavisMP) and hashtags (e.g. #brexit) are not contained in the Wikipedia pages. Therefore, the PMI and LSA-based metrics, using Wikipedia as the external resource, have limitations when evaluating the coherence of topics from tweets since

[12]This topic is generated from tweets posted in July and August 2016 in the UK. During this period of time, the referendum of **Br**itish **Exit** from the European Union (shorted as Brexit) just happened and Theresa May became Prime Minister of the UK on 11/07/2016. Meanwhile, the 2016 Summer Olympics was held in Brazil.

they are not likely to capture the semantic similarities of the newly emerged words on Twitter, such as these mentions and hashtags. For example, "#brexit" should be semantically similar to "@DavidDavisMP" (Twitter handle of Mr David Davis) as David Davis was the Secretary of State for Exiting the European Union.

In this section, we have reviewed several existing topic coherence metrics. All these metrics were initially proposed for evaluating the quality of topics generated from a normal text corpora, such as newspapers. However, it is not clear how well they work for topics generated from tweets. We mentioned that two promising metrics, namely the LSA and PMI-based metrics calculated using Wikipedia pages as an external resource, have limitations when applied on Twitter data since the Wikipedia pages do not necessarily cover all words encountered in tweets (e.g. hashtags). Instead, we propose new topic coherence metrics that are tailored to Twitter, which allow to evaluate the topic modelling approaches on Twitter data. In Chapter 4, we first evaluate the existing baseline topic coherence metrics for Twitter data, i.e. metrics U, V, B and metrics based on LCH, JCN, LESK, PMI and LSA. Then, we evaluate our tailored Twitter topic coherence metrics in comparison with these baseline metrics.

3.4 Twitter User Community Classification

We aim to analyse political events on Twitter by addressing 'who' said 'what' and'when'. While 'what' and 'when' can be addressed by an effective Twitter topic modelling approach taking into account the time dimension of Twitter, we aim to identify the 'who' by classifying users into communities. In this thesis, a community is a group of Twitter users having the same profession (e.g. a business elite, academic or media person) or supporting the same candidate or election campaign. To conduct the Twitter user community classification task, it is first important to obtain ground-truth data for training a user community classifier. Then, we can investigate how to effectively classify Twitter users into communities. Therefore, in this section, we first review the existing work in generating ground-truth data including through manual labelling and automatic labelling approaches (Sections 3.4.1 and 3.4.2). We then review the related work about Twitter user community classification approaches (Section 3.4.3). Note that a group of users with the same attribute (e.g. gender) is also treated as a community in the following review.

3.4.1 Manual Labelling Approaches

Typically, human annotators annotate an instance, in this case a Twitter user, to produce a label by examining the content of the data instance (e.g. their tweets) or the other relevant resources (e.g. the users' home pages and the user's profiles in other social media networks). To analyse agreement among human annotators, or to increase confidence in the accuracy of the obtained labels, some experimental designs obtain several annotations for every given instance. The instance is assigned a label, when this label is agreed by a majority of human annotators. Such a manual labelling method is widely used in the literature. For example, in a classification task of Twitter users' ages, the ground-truth labels (e.g. "young" and "junior") can be assigned to Twitter users by manually checking the LinkedIn[13] profiles and/or the homepages of these users. On the other hand, Al Zamal et al. (2012) used a Twitter dataset, where a user is assigned as "male" or "female". To confirm the correctness of these labels, human annotators compared Twitter users' full names to a record of newborn boys/girls' names obtained from a government body. In McDonald et al. (2014), human annotators identified whether a document was sensitive by reading the details of documents. To obtain such annotations efficiently, crowdsourcing platforms, e.g. CrowdFlower[14] and MTurk[15], are increasingly used to generate labels for data instances. For example, to quickly generate ground-truth data for query classification, McCreadie et al. (2010) showed how to make these workers[16] generate high-quality labels by presenting these workers with more relevant content. Similarly, Chen et al. (2015) asked crowdsourcing workers to identify the ethnicity, gender and age of Twitter users by showing them the names, profile images, self-descriptions, and tweets of Twitter users. Manual labelling can be time-consuming and expensive, especially when a large ground-truth dataset for training a reliable classifier is needed (Banko and Brill, 2001).

In this thesis, we aim to generate ground-truth data for training a user community classifier during political events. Since there are more and more political events happening on Twitter, it can be expensive to obtain the ground-truth community labels using a manual labelling approach for emerging political events. We aim to propose effective ground-truth generation approaches, which can be easily deployed and adopted for different political events. This leads us to automatic ground-truth generation approaches. Next, we review the existing automatic ground-truth generation approaches. In Chapter 6, we will show that such a manual ground-truth labelling approach is unnecessary when we can automatically generate good quality labels.

[13] `https://www.linkedin.com`
[14] `https://www.crowdflower.com`
[15] `https://www.mturk.com`
[16] A crowdsourcing worker is a human annotator in the crowdsourcing platform, e.g. CrowdFlower.

3.4.2 Automatic Labelling Approaches

In an automatic labelling approach, some manually pre-defined rules are first set by human assessors. Then, a ground-truth data can be automatically generated by using these pre-defined rules. These rules can correspond to symbols, hashtags, existing lists of users, and more. If a data instance matches the rules of a class, this instance is assigned to that class. For example, to generate ground-truth data for document classification, Damerau et al. (2004) assigned labels to documents by examining and considering the labels of their closest pre-defined documents, where these pre-defined documents were already labelled. Specifically, the nearest neighbour algorithm was applied to find the closest pre-defined document for each unlabelled document. Then, the unlabelled document was assigned a label, which is the same the as the label of the chosen labelled document. Read (2005) and Go et al. (2009) obtained training data for sentiment analysis by identifying whether a tweet contains relevant emotion symbols. For example, if a tweet contains ':-)', this tweet is likely to be positive. The labelled tweets were applied to conduct the sentiment classification task. Meanwhile, to identify whether a word corresponds to a technical terminology, Judea et al. (2014) used a set of pre-defined POS patterns to automatically identify the technique words in a document. If the POS tags of a sentence matched the pre-defined POS patterns, the noun of the sentence was chosen as a candidate word of the technique. Apart from these pre-defined rules, it is also popular for social scientists to use existing lists of people as ground-truth data. People can be grouped together using their professions (e.g. journalists), common interests or topics. For example, the Twitter public list[17] "UK MPs"[18] created by *Twitter Government* (Twitter handle @TwitterGov)[19] is a collection of UK members of parliament (MPs). Barberá et al. (2015) used the list of members of the Democratic and Republican parties (Twitter accounts) in the US to identify the political orientations of individuals. In these approaches, a Bayesian approach was used to infer the political orientations (i.e. latent variables) given the politician (members of parties in the US) followees as observations. To study the behaviours of "citizen journalists" on Twitter, Bagdouri and Oard (2015) proposed an automatic approach that used a seed list of journalists to identify a large number of "citizen journalists". Specifically, to find "citizen journalists", Bagdouri and Oard used the users' followee networks to check how users follow the journalists from the seed list and also applied the positive unlabelled learning to study the patterns between journalists in the seed list and candidate "citizen journalists". Similarly, Su et al. (2018) showed that the Twitter public lists can be used to generate a ground-truth data for classifying four user communities, i.e. business elites, academics, media and politics. However, Su et al. also found that the obtained ground-truth users generated using the

[17] https://help.twitter.com/en/using-twitter/twitter-lists
[18] https://twitter.com/twittergov/lists/uk-mps
[19] https://twitter.com/twittergov

public lists can also introduced noise in the classification and hence can cause difficulties when training user community classifiers. For example, the best F1 score obtained was 0.54.

Although these mentioned automatic labelling methods are promising to quickly generate ground-truth data, they cannot be adapted directly for our Twitter user community classification. For example, the mentioned pre-defined rules, such as emoticons, obviously cannot indicate the community affiliations of Twitter users. In particular, the ground-truth datasets used for training classifiers in different political events are different since the participated Twitter users and the discussed topics on Twitter are not the same. The pre-defined rules might not be easy to set in order to effectively labelled Twitter users into communities. Meanwhile, we show that the pre-defined lists can generate ground-truth data to classify the professions of Twitter users. However, it can have limitations. First, the Twitter users in the pre-defined lists are a small group of people who share similar interests or are interested in topics. They cannot represent a community in general. For example, in Bagdouri and Oard (2015), a Twitter public list of BBC journalists was used to generate ground-truth data for identifying a journalist community. However, these chosen journalists are different from a "citizen journalist" on Twitter, who is not a real journalist but shares stories and news. Second, the data obtained from the pre-defined lists can be small. For example, Su et al. obtained approximately 1000 users for each community, which might not be enough to train an effective user community classifier. To identify 'who' participated a political event on Twitter, it is important to investigate how to automatically generate ground-truth data suitable for the user community classification so as to develop user community classifiers for many political events. We aim to propose effective ground-truth generation approaches for classifying Twitter users into communities. Such approaches can be easy to apply and to deploy user community classifiers for emerging political events. Such trained classifiers can assist social scientists to examine the connections among communities with different political orientation and professions during a political event on Twitter.

3.4.3 Existing User Community Classification Approaches

In this section, we discuss the related work about Twitter user community classification in terms of classification task (Section 3.4.3.1) and the used features (Section 3.4.3.2).

3.4.3.1 Classification Tasks

Table 3.1 lists prior work in the classification of Twitter users into different communities. A community can be a group of people sharing the same age, gender, ethnicity or even the

same political orientation. Rustagi et al. (2009) classified the ages and genders of bloggers (users in `myspace.com`). Such a classifier was applied to analyse the styles of the posted words and sentences by bloggers in different ages and genders, which can then be used in an information retrieval system when matching documents. On the other hand, the ages and genders of Twitter users have also been studied. For example, Rao et al. (2010); Al Zamal et al. (2012); Vicente et al. (2019) have all investigated how to effectively classify the attributes (e.g. age and gender) of Twitter users, since such attribute information from users can be used when providing personalized services. Moreover, classifying the ethnicities of Twitter users is popular (such as in Pennacchiotti and Popescu, 2011; Culotta et al., 2015), since automatically identification of users' ethnicities can help to study the differences of linguistic among ethnicities. In addition, prior work has also studied how to classify the political orientations of users. For example, Thomas et al. (2006) used the congressional transcripts to determine whether a congressman supports a legislation, which can help people understand and analyse politically oriented documents. Due to the popularity of social media network, many work has attempted to classify Twitter users in terms of their political ideologies (i.e. left or right wing) (e.g. in Rao et al., 2010; Papadimitriou et al., 2000; Al Zamal et al., 2012). Indeed, studying the political orientations of Twitter users can help social scientists to study the communication between different communities with different political orientations. However, Cohen and Ruths (2013) reported that classifying Twitter users' political orientation was not an easy task. Specifically, Cohen and Ruths argued that the Twitter users used in prior work (i.e. Rao et al., 2010; Papadimitriou et al., 2000; Al Zamal et al., 2012) were users who explicitly indicated their political orientations. Cohen and Ruths showed that the classifier should be designed to determine the political orientations of the "modest" Twitter users who did not clearly declare their political views. Meanwhile, it has become popular to analyse Twitter users' political orientations during a political event. For example, Zubiaga et al. (2017) studied how Twitter users voted (supporting or opposing) in the Scottish and Catalonia independence referenda by developing political orientation classifiers. Similarly, Yilmaz and Abul (2018) targeted the classification of users in the Turkish constitutional referendum. On the other hand, a community can also correspond to people with a similar profession, such as journalists and business elites. De Choudhury et al. (2012) classified whether a given Twitter user was a journalist/blogger or an ordinary individual. The trained classifier was used to identify 'who' participated the online conversations. In addition, Su et al. (2018) studied how to effectively classify Twitter users into four communities in terms of Twitter users' professions by cleaning the training data. Such a community classifier could help to study how communities influence each others, which is a common study in social science (e.g. Vaccari et al., 2013).

Table 3.1: The related work about different classification tasks on Twitter.

Classifiction Tasks	Related Work
Age	Rustagi et al. (2009); Rao et al. (2010); Al Zamal et al. (2012); Morgan-Lopez et al. (2017)
Gender	Rustagi et al. (2009); Burger et al. (2011); Rao et al. (2010); Al Zamal et al. (2012); Wood-Doughty et al. (2018); Vicente et al. (2019)
Ethnicity	Rao et al. (2010); Pennacchiotti and Popescu (2011); Culotta et al. (2015); Wood-Doughty et al. (2018)
Political Orientation	Thomas et al. (2006); Rao et al. (2010); Pennacchiotti and Popescu (2011); Conover et al. (2011a); Al Zamal et al. (2012); Cohen and Ruths (2013); Barberá (2016); Zubiaga et al. (2017); Yilmaz and Abul (2018)
Professions	De Choudhury et al. (2012); Su et al. (2018); Aletras and Chamberlain (2018)

In this thesis, we aim to classify the political orientations of Twitter users during a political event, where the political orientation can be the voting preferences (e.g. "Yes" and "No") in a referendum or candidate voting preferences (e.g. "Donald Trump" or "Hillary Clinton") in an election. There are two main differences between our classification task and the tasks of the aforementioned existing work. First, we focus on classifying the political orientations of Twitter users during political events rather than their ideologies (left or right wing). A Twitter user can have different political orientations for different political events while the ideology of a Twitter might not change much. Second, since we aim to develop classifiers for different political events, the training data for different political events can be different. On the other hand, the training data used to train the classifiers in terms ideology in prior work can be the same (e.g. in Rao et al., 2010; Pennacchiotti and Popescu, 2011). Therefore, it can be more challenging to obtain ground-truth for our user community classification task. This again explains why we study the automatic ground-truth generation approach. Meanwhile, to assist social scientists to understand the connections among different communities, we also aim to classify Twitter users into communities in terms of their professions. As mentioned in Cohen and Ruths (2013), it is important to identify the community affiliations of the "modest" Twitter users. Such Twitter users are not significant figures and do not explicitly indicate their community affiliations. Since the majority of Twitter users belong to this type of users, it can be more useful to classify the community affiliations of the "modest" Twitter users and study their connections. In the next section, we give more details of the features used in these user community classification work.

3.4.3.2 Features used in User Classification

In this section, we review the features used for Twitter user classification (not only limited to user community classification). There are various features used in the user classification on

Table 3.2: The related classification work using different features on Twitter.

Features	Related Work
Word	Rustagi et al. (2009); Pennacchiotti and Popescu (2011); Lee et al. (2011); Burger et al. (2011); Conover et al. (2011a); Al Zamal et al. (2012); Cohen and Ruths (2013); Morgan-Lopez et al. (2017); Yilmaz and Abul (2018); Vicente et al. (2019)
n-grams	Rao et al. (2010); Al Zamal et al. (2012); Morgan-Lopez et al. (2017); Vicente et al. (2019)
Network	Rao et al. (2010); Pennacchiotti and Popescu (2011); Lee et al. (2011); Al Zamal et al. (2012); De Choudhury et al. (2012); Barber (2015); Bagdouri and Oard (2015); Hussain and Islam (2016); Aletras and Chamberlain (2018); Vicente et al. (2019)
User profile	Pennacchiotti and Popescu (2011); Burger et al. (2011); Barberá (2016); Bagdouri and Oard (2015); Morgan-Lopez et al. (2017); Wood-Doughty et al. (2018); Vicente et al. (2019)
retweeting &replying	Pennacchiotti and Popescu (2011); Al Zamal et al. (2012); De Choudhury et al. (2012); Vicente et al. (2019)
Topics	Pennacchiotti and Popescu (2011); De Choudhury et al. (2012); Cohen and Ruths (2013); Yilmaz and Abul (2018); Aletras and Chamberlain (2018)

Twitter as shown in Table 3.2[20]. Among these features, word features are widely used (see the first row in Table 3.2). Twitter users can post many tweets. By distinguishing the usage of words in the tweets, the community affiliations of Twitter users can be identified. For example, young people like to use newly emerging slang words on social media networks while old people tend to use more formal words and therefore the word features can be used to identify users' genders (Rustagi et al., 2009). Similarly, a male likes to use words *game* and *software* while a female favours words *cute* and *shopping*. Word features have been reported to work effectively to classify the genders of users (Morgan-Lopez et al., 2017; Vicente et al., 2019). In addition, by using word features, Pennacchiotti and Popescu (2011) obtained a F1 score of 0.808 when classifying the political orientations of Twitter users. In a topic classification task, Morgan-Lopez et al. (2017) showed that a trained classifier using word features only can have an accuracy score of 0.72 when classifying Twitter users into three age groups. Yilmaz and Abul (2018) even obtained an accuracy score of 0.88 in classification task of Twitter users' political orientations, where SVM was applied using word feature only. Indeed, word features are effective for the user community classification. Similarly to the use of word features, word n-grams are also widely used in many user classification work, such as the classification of the political orientation in Al Zamal et al. (2012). On the other hand, a user can follow the other users on Twitter, which can form a follower network. Since users might follow other users if they share the same interest or profession, these features can be useful in user classification. Al Zamal et al. (2012) used Twitter users' friends as features when classifying Twitter users' attributes. For example, the top 10 friends

[20]The cited work is the same as that in Table 3.1.

(assessed by the number of their followers) of Twitter users were used to classify their ages. If the top 10 friends were mostly popular singers, it is likely that the age of a Twitter user is low. Moreover, Pennacchiotti and Popescu (2011) showed that the user profile features, such as their self-descriptions and locations, can improve the performance of a classifier when identifying the ethnicity. Wood-Doughty et al. (2018) proposed a convolutional neural network model to predict users' genders and ethnicities using their screen names, where names were transformed to embedding vectors. Wood-Doughty et al. showed that the trained convolutional network model performed better than SVM using bag-of-words. A user profile can contain their national identity. Zubiaga et al. (2017) showed that Twitter users voted differently from the local people in a country during a referendum if the users do not have a national identity of this country. Hence, the national identity can be used to identify users' votes. A user can also make actions such as retweeting, replying, etc. These features can be useful to identify users' attributes similar to the network features. For example, a journalist can retweet tweets posted by news outlets and therefore can be used to classify whether a Twitter user is journalist (e.g. De Choudhury et al., 2012). Aletras and Chamberlain (2018) showed that the embedding vectors can be learnt from Twitter users' follower network and can be used to effectively classify users' professions. Specifically, Aletras and Chamberlain demonstrated that the Twitter users with the same profession can follow each other and therefore they will have similar embedding vectors.

In addition, it has been reported that topic[21] features can help a classifier to effectively identify users' attributes. For example. Pennacchiotti and Popescu (2011) used the labels of topics (e.g. 'music' and 'politics') as features to classify users' ethnicities and political orientations. Since Twitter users in different ethnicities can have different cultures, their discussed topics can be different and therefore can be used as features. As introduced in Section 2.2.1, a document has a distribution over topics in topic modelling. Similarly, a user can have also a distribution of topics inferred from their tweets. Hence, the topic distribution of a user can be used as features in a user community classification task, such as in a classification task of classifying users' professions (De Choudhury et al., 2012). Instead of using LDA, Preoţiuc-Pietro et al. (2015) applied Normalised Pointwise Mutual Information to create clusters of words, which were considered as topics. Then words were associated with topics, which were used in the profession classification. This approach was also used in Aletras and Chamberlain (2018) for classifying the profession.

In this thesis, we choose to use words as features in our user community classification task. There are three reasons. First, we show that word features are effective for the user community classification in the literature (e.g. in Rustagi et al., 2009; Pennacchiotti and

[21]These topics are generated using a topic modelling approach, such as LDA.

Popescu, 2011; Morgan-Lopez et al., 2017; Yilmaz and Abul, 2018). Second, we do not use the other aforementioned features, such as users' network, since they are time-consuming to obtain[22]. There can be many Twitter users participating in many political events, we aim to develop our classifiers using common features (i.e. words), which can be easily generalised across many user community classifications tasks for different political events. As mentioned in Section 3.4.3.1, the community classification for political events is a challenging task. We start with word features and leave the use of the other features as future work. On the other hand, we have shown that topics were used as features in the literature (e.g. used in Cohen and Ruths, 2013; Yilmaz and Abul, 2018; Aletras and Chamberlain, 2018). During a political event, there can be many topics related to the political event discussed by different Twitter users. Therefore, the discussed topics during a political event could be intuitively used to identify the community affiliations of Twitter users. In Chapter 6, we further investigate the usefulness of topic features in our user community classification task.

We have introduced several commonly used classification approaches in Section 2.4. These approaches have been widely used in the aforementioned user classification work. For example, Al Zamal et al. (2012) applied SVM to classify Twitter users' attributes and reported it can provide a good performance. On the other hand, deep learning neural networks models, such as convolutional networks, have been applied in the user community classification task. For instance, Benton et al. (2016) proposed to use neural networks to learn low dimension representations of Twitter users from their posted tweets, which can be used to predict users' attributes. Similarly, Aletras and Chamberlain (2018) proposed a skip-gram model to learn users' representations from their followee networks in order to classify users in terms of their occupations and incomes. Wood-Doughty et al. (2018) applied convolutional and recurrent neural networks to identify users' genders and ethnicities using users' names and Twitter screen names as inputs. In this thesis, we aim to automatically generate ground-truth data for training a Twitter user community classifier for a political event on Twitter (introduced in Section 3.4.2). We will apply the commonly used classifiers as baselines to identify whether the trained classifier (using the automatically generated ground-truth data) can effectively classify Twitter users into communities. We will apply Multi-layer Perception as one baseline, however, we do not involve the other aforementioned deep learning models in our classification work as they are not the focus of our research.

[22]The Twitter REST API (`https://dev.twitter.com/rest`) has to be intensively used to obtain these features (e.g. user profiles, follower network, etc.), and therefore one can quickly run up against Twitter imposed rate limits for that API.

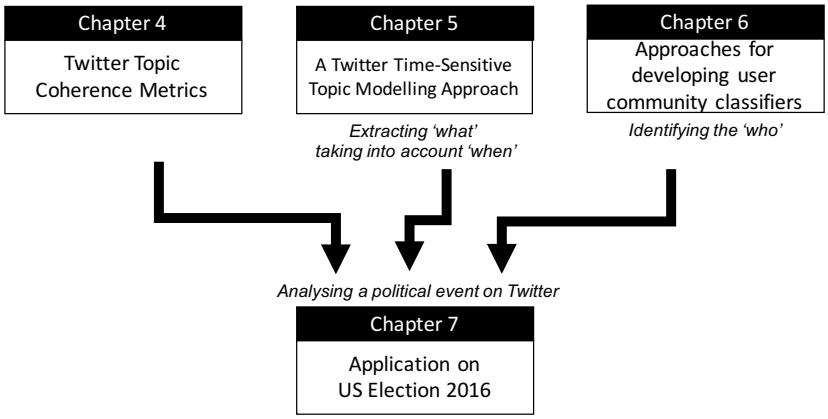

Figure 3.1: The approaches proposed in this thesis.

3.5 Overview of our Approaches

In the previous sections, we have reviewed the existing literature about topic modelling, topic coherence evaluation and user community classification for Twitter. We also discussed the main limitations of the existing work. In this section, we provide an overview of the approaches we will propose to address these limitations and to support the analysis of political events on Twitter. Figure 3.1 summarises our proposed approaches and their corresponding chapters.

First, we propose a novel Twitter time-sensitive topic modelling approach to generate topics with a high coherence (see Chapter 5). Since there can be many topics discussed on Twitter during a political event, it is important to integrate the time dimension of tweets in the topic modelling process so as to distinguish the discussed topics and improve their coherence. ToT (see Section 3.2.2.3) does integrate time. However, it was not initially designed for tweets and does not adequately control the importance of time with respect to the words typically used in topic modelling (see Section 3.2.2.3). Instead, we propose to integrate the time dimension of tweets using the Variational Bayesian (VB) implementation approach, where the importance of time and words can be balanced. Moreover, since the VB implementation approach can be efficient for large corpora (Hoffman et al., 2010), it seems particularly suitable to deal with the increasing volume of tweets posted during a political event.

Second, we propose novel semantic similarity-based metrics for evaluating topics generated from tweets. In Section 3.3.2, we argued that the Wikipedia pages do not have a good coverage of words occurring in tweets including hashtags. Hence, metrics that use Wikipedia as their external resource might not be effective on Twitter. Instead, we propose to use a Twitter background data as an external resource to estimate the coherence of topics

generated from tweets. Since word embeddings can effectively capture the semantic similarities between words (Mikolov et al., 2013b), we also propose a new coherence metric based on word embeddings (see Chapter 4). We show the effectiveness of these metrics though two large user studies.

Third, we propose automatic ground-truth generation approaches for training user community classification classifiers. As mentioned in Sections 3.4.1 and 3.4.2, the existing labelling approaches (e.g. the one using emoticons) cannot be directly applied in our user community classification. For example, these approaches have been proposed for other classification tasks and will not necessarily generalise to our user community classification task. We propose two ground-truth generation approaches for training and developing user community classifiers. Our two approaches are based on hashtags and DBpedia entities, respectively. For example, the first approach leverages hashtags, which are widely used by Twitter users during a political event (e.g. "#YesScot" and "#NoThanks" in the Scottish Independence Referendum 2014, see Brigadir et al., 2015). Our second approach makes use of DBpedia entities to associate the Twitter users with their professional communities. Such entities are related to users' professions and therefore can indicate their professional communities, e.g. entity "professor". Moreover, we mentioned in Section 3.4.3.2 that the topic features can be particularly useful for developing effective user community classifiers. Therefore, we also propose a tailored classifier to Twitter data, which classifies users by taking into account the different topics they discussed during a political event (see Chapter 6).

Finally, to show the effectiveness and the generalisation of our proposed approaches, we analyse the US Election 2016 event (see Chapter 7) with the help of a social scientist. Specifically, we apply our automatic ground-truth generation approach for training a classifier that identifies the communities, which supported the two presidential candidates (i.e. identifying the 'who'). We also apply our time-sensitive topic modelling approach to extract 'what' topics have been discussed and 'when' (i.e. the time dimension). We evaluate our generated topics by using our proposed Twitter coherence metrics and present social scientists with the coherent community-related topics for analysis.

3.6 Conclusions

In this section, we first reviewed the existing work about topic modelling approach in improving the coherence of topics generated from Twitter data (c.f. Section 3.2.2). We described the limitations of topic modelling approaches when extracting topics from a political event on Twitter. For example, some approaches (e.g. ToT) are not designed approaches for tweets while some approaches (e.g. topic modelling with word embeddings) do not deal with the

time dimension. We discussed that it was important to study the time dimension of tweets so as to capture the dynamics of conversations during a political event on Twitter. Second, we reviewed existing metrics that automatically evaluated the coherence of the generated topics. However, these existing topic coherence metrics were designed for newspaper articles. It is not clear how they perform for topics generated from Twitter data. In addition, the words used in a normal text corpus can be different from tweets (see Section 3.2.1). Indeed, we explained that the existing coherence metrics have limitations when applied on Twitter data, i.e. they might not effectively evaluate the coherence of topics from tweets. Third, we reviewed the existing work about ground-truth generation approaches for training user community classifiers. We discussed that the existing automatic labelling approaches have limitations for our user community classification task. For example, the used pre-defined rules cannot indicate the political orientations of Twitter users during a political event. Meanwhile, we mentioned that topic features can be useful for the user community classification task and that it is useful to investigate how topics differentiate Twitter users in different communities so as to develop an effective community classifier. To overcome these mentioned limitations, we introduced in Section 3.5, an overview of our proposals in this thesis, covering a time-sensitive Twitter topic modelling approach, new tailored topical coherence metrics and novel user community classifiers including two ground-truth generation approaches.

In the next chapter, we first investigate new Twitter topic coherence metrics, since such metrics are needed to evaluate the performance of a given topic modelling approach.

Chapter 4

Measuring Topic Coherence

4.1 Introduction

In the previous chapter, we have discussed the topic coherence metrics, which are used to evaluate the coherence quality of the generated topics by calculating the coherence scores of these topics. We have introduced several existing topic coherence metrics that evaluate the topics generated from normal text corpora. However, Twitter corpora are different from the normal text corpora (c.f. Section 3.2.1). It is not clear how the existing topic coherence metrics perform for the topics generated from Twitter data, which we call tweet topics. To effectively evaluate the coherence of the tweet topics, it is necessary to propose a tailored topic coherence metric for Twitter data. Hence, in this chapter, we investigate approaches to measure the coherence of the tweet topics. As discussed in Section 2.2, a topic modelling approach generates a topic model containing K topics[1]. We also propose to assess the global coherence of a generated topic model.

In this chapter, we first examine various topic coherence metrics and the extent to which they align with human coherence judgements. Specifically, we adapt several existing coherence metrics initially proposed for evaluating the coherence of topics generated from traditional text corpora to Twitter data. We then propose new coherence metrics that incorporate improvements tailored to tweet topics. We show that these proposed metrics align better with human coherence judgements than various coherence metrics from the literature. Next, using the best-proposed coherence metric, we propose a `coherence at n` metric to assess the global coherence of a topic model by averaging the coherence scores of the top n ranked topics in a given topic model. We conduct a large-scale experiment to show the

[1] K is the number of topics, which is a parameter of a topic modelling approach.

effectiveness of the `coherence at n` metric compared to the commonly used average coherence score of K topics. The remainder of the chapter is organised as follows:

- Section 4.2 first describes details of the existing baseline topic coherence metrics.
- Section 4.3 introduces our proposed topic coherence metrics.
- Section 4.4 explains the methodology we use to compare the effectiveness of topic coherence metrics through a user study.
- In Section 4.5, we evaluate our proposed topic coherence metrics, in comparison to several metrics adapted from the literature.
- Section 4.6 introduces the global `coherence at n` metric and evaluates its effectiveness.
- Section 4.7 provides some concluding remarks for this chapter.

4.2 Baseline Topic Coherence Metrics

In Section 3.3, we have discussed several topic coherence metrics including metrics based on statistical analysis and metrics based on semantic similarity. These metrics are used as baselines in this chapter. For the metrics based on statistical analysis, we apply the metrics U, V and B (c.f. Section 3.3.1) as baselines in this chapter. For the semantic similarity-based metrics, we use as baselines the metrics based on LCH, JCN, LESK, LSA and PMI, which were introduced in Section 3.3.2. We denote these metrics by LCH, JCN, LESK, W-LSA and W-PMI (where 'W' denotes the use of Wikipedia pages as an external resource), respectively (see Table 4.1). For all the semantic similarity-based metrics, we choose to use the top-n ranked words[2] to calculate the coherence of topics using Equation (4.1):

$$Coherence(topic) = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j=i+1}^{n} f_{SS}(w_i, w_j) \tag{4.1}$$

where f_{ss} is the semantic similarity function. We use the top 10 words (i.e. set n to 10) to represent the content of a generated topic since this setting is also used by Newman et al. (2009, 2010)

In total, we apply 8 existing topic coherence metrics from the literature, listed in Table 4.1. As discussed in Section 3.3.2, all the semantic similarity-based metrics (i.e. metric

[2]Ranked by their probabilities in the topic term distribution.

numbered 4-8 in Table 4.1) require an external resource to compute the coherence scores. For example, W-LSA and W-PMI leverage Wikipedia pages as an external resource to calculate the semantic similarities of words using LSA and PMI, respectively. LCH, JCN and LESK use WordNet (Miller, 1998) as their external resource. All the semantic similarity-based metrics use Equation (4.1) to evaluate the coherence of topics. The difference between these metrics is that their used f_{ss} functions are different. For example, the f_{ss} function in W-LSA computes the semantic similarities of words using the trained LSA model. The f_{ss} function in LCH calculates the semantic similarities of words by using WordNet (additional details are provided in Section 3.3.2).

Table 4.1: The baseline topic coherence metrics introduced in Chapter 3.

	Metrics	Description
1-3	U, V & B	Comparing distances between topics and three meaningless topics (AlSumait et al., 2009).
4	LCH	f_{ss} is implemented by using LCH (Leacock and Chodorow, 1998).
5	JCN	f_{ss} is implemented by using JCN (Jiang and Conrath, 1997).
6	LESK	f_{ss} is implemented by using LESK (Lesk, 1986).
7	W-LSA	f_{ss} is implemented by the trained LSA model from Wikipedia pages (Newman et al., 2010).
8	W-PMI	f_{ss} is implemented by the trained PMI data from Wikipedia pages (Newman et al., 2010).

4.3 Twitter-specific Topic Coherence Metrics

Since the semantic similarity-based coherence metrics are reported to perform promisingly (see Section 3.3.2), we choose to develop our Twitter topic coherence metrics as semantic similarity-based metrics. In this chapter, we introduce two new methods to improve the semantic similarity-based metrics when assessing the coherence of tweet topics. For each method, we propose several possible variants.

4.3.1 Using an External Resource for Coherence Metrics

When a human assesses the coherence of topics, they commonly read the top-ranked words of a topic and then try to connect these words by considering their semantic similarity. Specifically, below we list a topic example that is extracted from tweets related to the referendum of Britain exiting from the European Union (commonly called Brexit) posted in July 2016:

> #theresamaypm #iamteamgb #rio #brexit minister people speech @harryslast-stand michelle britain

The word "#theresamaypm" is semantically similar to the word "minister". The metrics based on semantic similarity (e.g. metrics W-PMI and W-LSA, evaluating the topic coherence using Wikipedia pages as an external resource) are supposed to capture the similarity between these two words. However, the words "#theresamaypm", "#iamteamgb", "#rio and "#brexit" on Twitter do not appear in Wikipedia pages, which makes it difficult to compute the semantic similarities of words and thus the coherence of topics.

To capture all of the possible words encountered in tweet topics, we propose semantic similarity-based coherence metrics using a Twitter background dataset as an external resource. The Twitter background data contains 1% random tweets[3] from Twitter and hence records various abbreviations and hashtags. We argue that the semantic similarity of tweet topics can be better computed using background data obtained from Twitter. Accordingly, we postulate that such metrics can better align with human judgements. Similar to the PMI-based and LSA-based metrics using Wikipedia pages as an external resource (see W-PMI and W-LSA in Table 4.1), we propose two metrics: a PMI-based metric and a LSA-based metric using a Twitter background dataset as an external resource, which we denote by T-PMI and T-LSA, respectively. These two metrics leverage the PMI scores and a LSA model trained using the Twitter background data. We provide more details about how we collect this Twitter background dataset in Section 4.5.

4.3.2 Word Embedding-based Metrics

Recently, a newly proposed Neural Networks-generated word embedding (WE) model is reported to produce more effective word representations than those by LSA (Mikolov et al., 2013a,b,c). Using WE word representations has been shown to successfully improve the performance of classification (Lebret and Collobert, 2015) and machine translation (Zou et al., 2013) tasks. Compared to a LSA model (c.f. Section 3.3.1), a WE model is much faster to train (Lebret and Collobert, 2015) since the word embeddings training technique, skip-gram or CBOW (see (Mikolov et al., 2013a)), is highly parallelisable. The WE model has also been reported to effectively capture the semantic similarities between words (e.g. in Pennington et al., 2014; Kenter and De Rijke, 2015). Therefore, we propose new coherence metrics that leverage word embeddings to calculate the coherence scores of topics. Similar to the LSA model, a word in a WE model is represented as a vector, V_{m_j}, which is pre-trained using an external resource. A WE-based metric computes the coherence score of topics by using the equations shown as follows:

[3]This background dataset can be obtained using the Twitter public Streaming API.

$$f_{SS}(w_i, w_j) = cosine(V_{w_i}, V_{w_j})$$
$$Coherence(topic) = \frac{1}{\binom{n}{2}} \sum_{i=1}^{n} \sum_{j=i+1}^{n} f_{SS}(w_i, w_j) \tag{4.2}$$

where the word semantic similarity is computed by using cosine in f_{SS}. If the words within a topic are semantically similar (i.e. a high cosine score), the topic is coherent and can be more easily interpreted. To fully assess the performance of WE-based metrics, we use the word embedding models both trained using Wikipedia pages (denoted W-WE) and using a Twitter background dataset (denoted T-WE). When pre-training a WE model, its parameters, the size of the context window and the size of the dimension of embedding vectors, can impact the quality of the WE-based metrics. The size of the context window indicates the number of adjacent words that are used to determine the context of a word while the size of the dimension of embedding vectors is the number of dimensions of embedding vectors (Mikolov et al., 2013a). Therefore, we vary these parameters and examine their impact. The rest of the experimental setup is reported in Section 4.5.

Recently, deep contextual word embeddings have been proposed to solve the problems in the field of natural language processing, such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018). We do not use these contextual word embeddings to evaluate the coherence of topics in this thesis, since it is not straightforward to calculate the semantic similarities of words in a topic using deep contextual word embeddings. We leave this as future work.

Table 4.2: Our proposed Twitter topic coherence metrics.

	Metric	Description
1	T-LSA	f_{ss} is implemented by the trained LSA space from a Twitter background data.
2	T-PMI	f_{ss} is implemented by the trained PMI data from a Twitter background data.
3	W-WE	f_{ss} is implemented by the trained WE space from Wikipedia.
4	T-WE	f_{ss} is implemented by the trained WE space from a Twitter background data.

So far, we have described 8 existing coherence metrics (see Table 4.1) and have introduced 4 proposed topic coherence metrics (listed in Table 4.2), i.e.T-LSA, T-PMI, W-WE and T-WE. In the next section, we introduce our methodology to evaluate the performance of the 4 proposed coherence metrics in comparison to the 8 existing coherence metrics using human judgements.

4.4 Metrics Comparison Methodology

In this section, we introduce the methodology we use to identify which topic coherence metric best aligns with the human judgements of topical coherence. We conduct a user study to obtain human judgements. However, it can be a challenging task for human assessors to produce graded coherence assessments of topics. Therefore, we use a pairwise preference user study to gather the human judgements. A similar method has been previously used to compare summarisation algorithms (Mackie et al., 2014) and the relevance of documents given a query (Carterette et al., 2008). In the following, Section 4.4.1 explains how we generate topic pairs. Section 4.4.2 describes how we conduct a pairwise crowdsourced user study to obtain the human judgements. We compare metrics to human judgements in terms of their agreement (see Section 4.4.3) and by the extent to which they rank the performance of topic modelling approaches similarly to human assessors (see Section 4.4.4).

4.4.1 Generating Topics

To perform the pairwise preference user study, we generate topic pairs using three topic modelling approaches: the classical Latent Dirichlet Allocation (LDA), Twitter Specific LDA (TLDA) and Pachinko Allocation (PAM) (see Sections 2.2 and 3.2.2). These three topic modelling approaches generate different topics[4], which makes it possible for human assessors to identify a better topic out of two topics generated using two different topic modelling approaches. These three topic modelling approaches also allow us to study the performance of different topic modelling approaches in terms of generating coherent topics.

Table 4.3: The *comparison units* of the three topic modelling approaches.

Comparison Unit	Topic Pairs in Unit
(1) Unit(LDA, TLDA)	Pairs(LDA→TLDA & TLDA→LDA)
(2) Unit(LDA, PAM)	Pairs(LDA→PAM & PAM→LDA)
(3) Unit(TLDA, PAM)	Pairs(TLDA→PAM & PAM→TLDA)

More specifically, we divide the comparison task into three *comparison units*: LDA vs. TLDA, LDA vs. PAM and TLDA vs. PAM. Each *comparison unit* consists of a certain number of topic pairs and each pair contains a topic from topic models T_1 and T_2, respectively (e.g. LDA vs. TLDA). To make the comparisons easier for human assessors, we only present similar topics in a pair. Specifically, each topic model has a set of candidate topics, and each topic is represented as a topic vector using its term distribution. We randomly select a certain number of topics from topic model T_1[5]. For each topic selected in T_1, we use

[4]For example, TLDA generates topics that do not contain many background words (c.f. Section 3.2.2.1).

[5]We do not use all the topics in a topic model due to the budget limit of our user study.

Equation (4.3) below to select the closest topic[6] in T_2 using cosine similarity[7]. The selected topic pairs are denoted as Pairs($T_1 \rightarrow T_2$). Moreover, to reduce the bias on model T_i, we also generate the same number of topic pairs, i.e. Pairs($T_2 \rightarrow T_1$), for Unit(T_1,T_2). Therefore, every *comparison unit* has a set of topic pairs shown in Table 4.3.

$$closest(topic_j^{T_1}) = argmin_{i<K}\ (1 - cosine(V_{topic_j^{T_1}}, V_{topic_i^{T_2}})) \tag{4.3}$$

4.4.2 User Study

In this section, we describe our conducted pairwise user study in order to obtain human judgements on topic preferences. As described in Section 4.4.1, the comparison task is divided into three *comparison units* and each *comparison unit* has a certain number of topic pairs. We ask human assessors to conduct a pairwise preference evaluation, and we use the obtained human assessors' preferences of topics from the topic models to identify the agreement (see Section 4.4.3) and rank the three topic modelling approaches (see Section 4.4.4). For collecting human judgements, we use the CrowdFlower[8] crowdsourcing platform.

Figure 4.1: The designed user interface on CrowdFlower for obtaining the topic coherence judgements.

For each topic pair in our three *comparison units*, we present a worker (i.e. a human) with the top 10 (discussed in Section 4.2) highly frequent words from the two topics (a topic pair, generated from two topic modelling approaches) along with their associated 3 most retweeted tweets, which are likely to represent the topic. A CrowdFlower worker is asked to choose the more coherent topic from two topics using these 10 words. The user interface of our user study is shown in Figures 4.1 and 4.2. To help the workers understand and finish the task, we provide guidelines that define a coherent topic as one that mixes fewer

[6]We only pair a topic with its closest topic to reduce the total number of topic pairs since the budget of our user study is limited.

[7]We use cosine similarity since it performs well on finding the closest topic in our preliminary experiment.

[8]`https://www.crowdflower.com`

Figure 4.2: The associated tweets for the two shown topics in our user study.

discussions/events and that can be interpreted easily. We instruct workers to consider: 1) the number of semantically similar words among the 10 shown words in a topic, 2) whether the 10 shown words imply multiple topics and 3) whether the 10 shown words provide more details about a discussion/event. If a decision cannot be made with these 10 words, a worker can then use the optional 3 associated tweets, shown in Figure 4.2. We provide two guidelines for using these tweets for assistance: 1) consider the number of the 10 shown words from a topic that can be reflected by the tweets and 2) consider the number of tweets that are related with the topic. After the workers make their choices, they are asked to select the reasons for their choices, as shown in Figure 4.1. The CrowdFlower workers are paid $0.05 for each judgement per topic pair. We gather 5 judgements[9] for each topic pair from 5 different workers. If no topic is preferred among the 5 judgements[10], more judgements are collected until a preferred topic is identified.

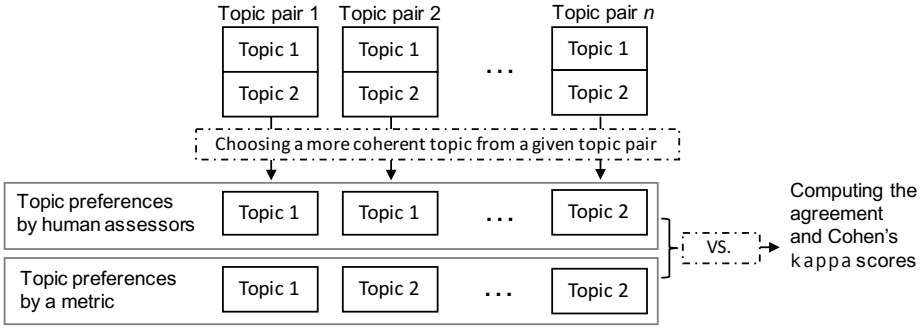

Figure 4.3: Agreement between metrics and human assessors on topical coherence.

4.4.3 Agreement between Metrics and Human Assessors

As mentioned in the previous section, we generate a set of topic pairs for the three *comparison units* (see Table 4.3). For example, Figure 4.3 illustrates a number of topic pairs that

[9]We argue that 5 is a reasonable number of judgements to determine the more coherent topic from the two given topics, since it indicates a clear majority vote.

[10]Users can choose the opinion of "No Preference" in our study.

will be assessed by both human assessors and a given metric. In the user study, each topic pair receives judgements from at least 5 human assessors. We assign a topic in a topic pair a fraction of the votes received. The topic that receives the majority of human votes is deemed to be the preferred topic as determined by humans (a topic preference, i.e. either Topic 1 or Topic 2 in Figure 4.3). On the other hand, a metric can also identify the topic with the higher coherence from a topic pair. Given a set of topic pairs, we first compute the number of topic pairs where a topic coherence metric can identify the same preferred topics (i.e. the "Topic preferences by a metric" in Figure 4.3) as the human assessors. We then calculate the proportion of these topic pairs with respect to the total number of topic pairs and denote it as the agreement score between the human assessors and the given topic coherence metric. We also report Cohen's $kappa$ agreement (Artstein and Poesio, 2008) between human assessors and each coherence metric. The results of the conducted user study are reported in Section 4.5.4.2.

4.4.4 Ranking of Topic Modelling Approaches

Aside from the agreement comparison, we also check whether a metric can rank the performance of the three used topic modelling approaches (i.e. LDA, TLDA and PAM) like human assessors. If a topic coherence metric can evaluate the coherence of topics, it can also differentiate the performance of the three topic modelling approaches in terms of topical coherence.

In a *comparison unit*, there are a set of topic pairs, where these topic pairs consist of two sets of topics generated using two topic modelling approaches. For example, Unit(LDA, TLDA) has topic pairs that contain two sets of topics generated using LDA and TLDA, respectively. For a topic pair, an automatic coherence metric computes the coherence scores of the two topics in this topic pair. Thus, for each *comparison unit*, we obtain two sets of coherence scores corresponding to the two topic modelling approaches. We then apply the Wilcoxon signed-rank test to calculate the significance level of the difference between the two sets of coherence scores. A topic modelling approach is better than another if the set of coherence scores of this topic modelling approach are significantly higher than that of another topic modelling approach. Therefore, for each *comparison unit*, an automatic coherence metric determines the better topic modelling approaches (e.g. LDA $>$ TLDA), which results in a ranking order of the three topic modelling approaches. For instance, given the preferences LDA>TLDA, LDA>PAM and TLDA>PAM, we can obtain the ranking order LDA(1^{st})>TLDA(2^{nd})>PAM(3^{rd}). However, while it is possible for the obtained preference results not to permit a ranking order – i.e. a Condorcet paradox such as TLDA>LDA, LDA>PAM & PAM>TLDA – we did not observe any such paradoxes in our experiments.

Similarly as above, we also obtain the ground-truth ranking order of the three topic modelling approaches using the topic coherence assessments obtained from human assessors. For example, the ground-truth ranking order is TLDA^{1st}>LDA^{2nd}>PAM^{3rd}. If a metric can also obtain the exact same order as that obtained from human assessors, we say that the ranking order of a metric **matches** the ranking order of human assessors. If a metric obtains a ranking order such as TLDA^{1st}>LDA$^{2nd/3rd}$>PAM$^{2nd/3rd}$, i.e. there are no significant performance differences between LDA and PAM, we say that the ranking order of this metric **partly matches** the ground-truth ranking order.

In summary, we have introduced the methodology that we use to evaluate the topic coherence metrics compared to the human judgements. In the next section, we evaluate our proposed 4 topic coherence metrics together with 8 existing baseline topic coherence metrics.

4.5 Evaluation of the Coherence Metrics

This section evaluates our proposed 4 topic coherence metrics (see Table 4.2) compared to the 8 existing coherence metrics (see Table 4.1) when assessing the coherence of tweet topics. Specifically, Section 4.5.1 describes the used Twitter datasets. We explain our experimental setup in Section 4.5.2 and list our research questions in Section 4.5.3. Section 4.5.4 reports the evaluation results of all the metrics.

4.5.1 Datasets

We describe the three datasets we use in our experiments:

Two datasets for topic modelling: In our experiments, we use two Twitter datasets to compare the topic coherence metrics. The first dataset we use consists of tweets posted by 2,853 journalists in the state of New York from 20/05/2015 to 19/08/2015, denoted as NYJ. To construct this dataset, we tracked the journalists' Twitter handles using the Twitter Streaming API[11]. We choose this dataset due to the high volume of topics discussed by journalists on Twitter. The second dataset contains tweets posted between 8pm-10pm on 02/04/2015, which is related to the first TV debate (denoted as TVD) during the UK General Election 2015. This dataset was collected by searching the TV debate-related hashtags and keywords (e.g. #TVDebate and #LeaderDebate) using the Twitter Streaming API[11]. We choose this dataset because social scientists want to understand what topics people discuss during a political event. Table 4.4 reports the details of these two datasets.

[11] `https://dev.twitter.com`

Note that the Twitter users in the TVD dataset are users who posted tweets about the TV Debate. On the other hand, the Twitter users in the NYJ dataset are professional journalists. These Twitter users usually post tweets containing less noise and more formal language. Hence, the NYJ dataset is much less noisy than the TVD dataset. The different nature of the two used Twitter datasets allow us to examine the performance of a given topic coherence metric across two types of datasets. In particular, an effective topic coherence metric should work equally effectively on NYJ as well as the more noisy TVD Twitter dataset.

Table 4.4: The details of the two used Twitter datasets for the study of the coherence metrics.

Name	Time Period	The number of users	The number of tweets
NYJ dataset	20/05/2015-19/08/2015	2,853	946,006
TVD dataset	8pm-10pm 02/04/2015	121,594	343,511

A Twitter background dataset: We use the Twitter public streaming API[11] to crawl a background Twitter dataset, which represents 1% random tweets crawled from 01/01/2015 to 30/06/2015. We remove stopwords, words occurring in less than 20 tweets, and the retweets. The remaining tweets (30,151,847) are used to pre-train the PMI data, the LSA and the WE models, which are used in our T-PMI, T-LSA and T-WE metrics, respectively. Additional details about our experimental setup are provided in the next section.

4.5.2 Experimental Setup

In this section, we first explain how we generate topics and topic pairs in Section 4.5.2.1. The user study quality control is described in Section 4.5.2.2. We explain our metrics setup in Section 4.5.2.3.

Table 4.5: The number of the *comparison units* on the two used Twitter dataset.

Comparison unit	NYJ dataset	TVB dataset	Total number per unit
(1) Unit(LDA, TLDA)	50 Pairs(LDA→TLDA) 50 Pairs(TLDA→LDA)	50 Pairs(LDA→TLDA) 50 Pairs(TLDA→LDA)	200
(2) Unit(LDA, PAM)	50 Pairs(LDA→PAM) 50 Pairs(PAM→LDA)	50 Pairs(LDA→PAM) 50 Pairs(PAM→LDA)	200
(2) Unit(TLDA, PAM)	50 Pairs(TLDA→PAM) 50 Pairs(PAM→TLDA)	50 Pairs(TLDA→PAM) 50 Pairs(PAM→TLDA)	200
Total number per dataset	300	300	600 (in total)

4.5.2.1 Generating Topics using the Topic Modelling Approaches

We use Mallet[12] and Twitter LDA[13] to deploy the three topic modelling approaches (i.e. LDA, TLDA and PAM) on the two datasets (described in Section 4.5.1). The LDA hyper-parameters α and β are set to $50/K$ and 0.01 respectively, which work effectively for most corpora (Steyvers and Griffiths, 2007). In TLDA, we follow Zhao et al. (2011b) and set γ to 20. We set the number of topics K to a higher number, 100, for the NYJ dataset as it contains many topics. The TVD dataset contains fewer topics, particularly as it took place only over a 2 hour period, and politicians were asked to respond to questions on specific themes and ideas[14]. Hence, we set K to 30 for the TVD dataset. Each topic modelling approach is ran 5 times for each of the two datasets. Therefore, for each topic modelling approach, we obtain 500 (5 times 100) topics in the NYJ dataset and 150 (5 times 30) topics in the TVD dataset. We use the methodology described in Section 4.4.1 to generate 100 topic pairs for each comparison unit as shown in Table 4.5. For example, for Unit(LDA,TLDA), we generate 50 topic pairs of Pairs(LDA→TLDA) and 50 topic pairs of Pairs(TLDA→LDA). In summary, we generate 200 topic pairs for each *comparison unit*. For each dataset, we have 300 topic pairs.

4.5.2.2 CrowdFlower Quality Control

To ensure the quality of the obtained CrowdFlower judgements, we use several quality control strategies. We provide a set of test questions, where for each question, workers are asked to decide a topic preference from a topic pair. The answers of the test questions are verified in advance. Moreover, the worker must have maintained 70% or more accuracy on the test questions in the task, otherwise their judgements are discarded. Only workers that pass the test are allowed to enter the task. For the NYJ dataset, we limit the workers' country to the US only since the NYJ dataset contains only tweets posted in the US. The TVD dataset contains topics that can be easily understood, and thus we set the workers' country to English speaking countries (i.e. the UK, the US and Canada.).

4.5.2.3 Metrics Setup

Table 4.6 lists the details of all the used metrics. Metrics (1)-(8) are the baseline metrics (also introduced in Table 4.1). The metrics U, V, B (i.e. Metrics (1)-(3)) do not require

[12] https://mallet.cs.umass.edu

[13] https://github.com/minghui/Twitter-LDA

[14] https://en.wikipedia.org/wiki/United_Kingdom_general_election_debates,_2015

Table 4.6: The details of all the used topic coherence metrics. We use k to denote a thousand, i.e. $117k$ means $117,000$.

	Metrics	Metric variants	External Resource	Description	Statistics
Baseline metrics	(1) U (2) V (3) B		No external resource	Statistical analysis-based metrics, see Section 3.3.1.	N/A
	(4) LCH (5) JCN (6) LESK		WordNet	Implemented using the *WordNet::Similarity* package.	$117k$ synonym sets
	(7) W-LSA (8) W-PMI		Wikipedia pages	The PMI data and LSA model are obtained from *SEMILAR* platform.	$1m$ vectors $179m$ word pairs
Proposed metrics	(9) T-LSA (10) T-PMI		Our Twitter background dataset	We obtain the PMI data and LSA model from our Twitter background dataset.	$609k$ vectors $354m$ word pairs
	(11) W-WE	(11) G-W-WE$^{w=10}_{d=200}$ (12) G-W-WE$^{w=10}_{d=300}$	Wikipedia pages	We use the pre-trained WE models generated from Wikipedia pages in $GloVe$.	$400k$ vectors
	(12) T-WE	(13) G-T-WE$_{d=100}$ (14) G-T-WE$_{d=200}$	Twitter data in $GloVe$	We use the pre-trained WE models generated from a Twitter dataset in $GloVe$.	$119k$ vectors
		(15) T-WE$^{w=1}_{d=200}$ (16) T-WE$^{w=1}_{d=500}$ (17) T-WE$^{w=3}_{d=200}$ (18) T-WE$^{w=3}_{d=500}$	Our Twitter background dataset	We train our WE models generated from our Twitter background dataset using different sizes of context window and different size of the dimensions of vectors.	$504k$ vectors

an external resource. They are implemented as in AlSumait et al. (2008). The LCH, JCN and LESK (i.e. Metrics (4)-(6) in Table 4.6) are implemented using the *WordNet::Similarity* package (Pedersen et al., 2004). These three metrics evaluate the coherence of topics using $117k$ sets of synonyms. To implement the baseline metrics W-LSA and W-PMI (Metrics (7)-(8) in Table 4.6), i.e. the metrics that use Wikipedia pages as their external resource, we use the pre-trained LSA model and the PMI data generated from the Wikipedia pages as provided in the *SEMILAR* platform[15].

Metrics (9)-(12) are our proposed metrics that either use word embeddings or that use our Twitter background dataset (or both). We train the LSA model and the PMI data using our Twitter background dataset, which allows to implement T-LSA (Metric (9)) and T-PMI (Metric (10)) containing $609k$ vectors and $354m$ word pairs, respectively. We do not tune the parameters in Metrics (1)-(10) since these metrics have been studied in prior work (e.g. in AlSumait et al., 2009; Newman et al., 2010)[16]. Hence, Metrics (1)-(10) only have one variant, i.e. themselves (also called Metric variants (1)-(10) in Table 4.6). In terms of metrics based on word embeddings (i.e. W-WE and T-WE), we tune the parameters of the size of context window (w) and the dimension size of the embedding vectors (d) so as to identify whether these parameters impact the performance of these metrics. The W-WE, Metric (11), has two variants (i.e. Metric variants (11) and (12)). These two variants are implemented using the pre-trained WE models generated from the Wikipedia pages in $GloVe$ (Pennington et al., 2014), where the pre-trained WE models (containing $400k$ vectors) use 10 as the context

[15] `http://semanticsimilarity.org`

[16] Note that Metrics (9) and (10) are our proposed metrics. We follow Newman et al. (2010) to obtain the LSA model and PMI data from Twitter data.

window size and {200, 300} as the dimension sizes of the vectors. Hence, we denote the Metric variants (11) and (12) as G-W-WE$^{w=10}_{d=200}$ and G-W-WE$^{w=10}_{d=300}$, respectively. Similarly, we also use the pre-trained WE models (containing $119k$ vectors) generated from a Twitter dataset[17] in $GloVe$, denoted as G-T-WE$_{d=200}$ and G-T-WE$_{d=300}$. We train our WE models using our Twitter background dataset (see Section 4.5.1) once we set w and d to $w = \{1, 3\}$ and $d = \{200, 500\}$, respectively. All the trained WE models have $504k$ vectors. We use these trained WE models to implement Metric variants (15)-(18), which all belong to the metric family T-WE, i.e. Metric (12) in Table 4.6.

We notice that the stemmed WE model has a better performance than that of the unstemmed model in our experiments. Hence, we stem the words in our 4 trained WE models (i.e. Metrics variants (15)-(18), T-WE$^{w=\{1,3\}}_{d=\{200,500\}}$). To summarise, in this section, we have defined 18 metric variants, including 10 new metric variants (see Table 4.6), which we will evaluate in the next section. For readability purposes, we call all the metric variants by their family metric name (e.g. T-WE$^{w=1}_{d=500}$ is a T-WE metric).

4.5.3 Research Questions

In the remainder of this section, we aim to answer the following research questions:

- **RQ1.** Which coherence metric best aligns with the human judgements?
- **RQ2.** Can the Twitter background dataset help coherence metrics to better align with the human judgements compared to metrics using the background from other external resources?
- **RQ3.** Which configuration (i.e. the context window size and the dimension size of the vectors) leads to the most effective T-WE metric variant?

4.5.4 User Study Results

In this section, we first report statistical information about our user study (Section 4.5.4.1). We then compare all the metrics to the human judgements in terms of their agreement (Section 4.5.4.2) and the ranking of the three topic modelling approaches (Section 4.5.4.3). Finally, we summarise the results in Section 4.5.4.4.

[17]This Twitter dataset is provided in `https://nlp.stanford.edu/projects/glove/`. However, the size of the used context window in the trained WE model in $GloVe$ is unknown.

4.5.4.1 User Study Statistics

In our user study, we obtain 1804 judgements from 77 different trusted workers and 1918 judgements from 91 workers for the NYJ and TVD datasets, respectively. In a topic pair, if a topic obtains more votes from the human assessors, we consider that this topic is preferred as a more coherent topic by the human assessors. Figure 4.4 shows the confidence distribution of 300 topic pairs for each dataset, where the confidence of a topic pair is the proportion of votes for the preferred topic. For the NYJ dataset, 67.7% of topic preferences obtain more than 80% confidence score[18], while this figure is 55.4% for the TVD dataset. This suggests that the human assessors have a high agreement[19] on most of the topic preferences. There are more topic pairs (67.7%) in the NYJ dataset have a high confidence (i.e. confidence $> 80\%$) than those (55.4%) in the TVD dataset, which suggests that the topics in the TVD dataset are indeed more difficult for the human assessors to interpret than the topics in the NYJ dataset (also discussed in Section 4.5.1).

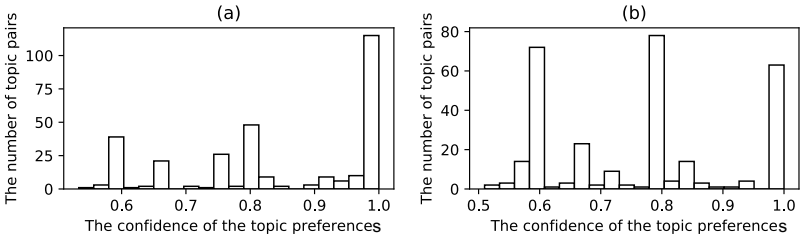

Figure 4.4: The confidence distribution of topic preferences from human judgements. (a) The NYJ dataset. (b) The TVD dataset.

4.5.4.2 Agreement between Metrics and Human Assessors

For each dataset, we have 300 topic preferences for 300 topic pairs (see Table 4.5) obtained from the human judgements. On the other hand, a metric also generates 300 topic preferences per dataset. We present both the agreement and Cohen's $kappa$ scores (c.f. Section 4.4.3) between the human-generated topic preferences and the topic preferences generated by each metric in Figures 4.5 and 4.6, respectively. Since there are three options ("Topic 1, "Topic 2 and "No Preference") in the topic preference task, the random baseline agreement rate is 33.3%.

All our proposed T-WE metrics, i.e. Metrics (15)-(18) in Figures 4.5 and 4.6, using the Twitter background dataset, have high and consistent agreement and Cohen's $kappa$ scores

[18]The preferred topic obtains 4 votes out of 5.

[19]Among 5 human assessors, if more than 4 assessors agree that topic A is more coherent than topic B in a topic pair, we say the agreement is high.

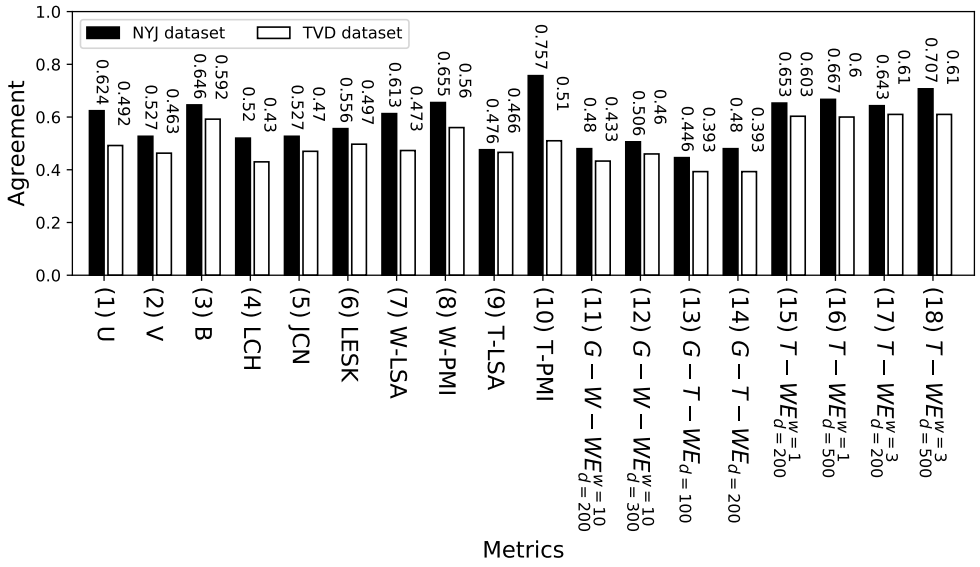

Figure 4.5: The topic preference agreement between the human judgements and the 18 metrics. Each bar in the figure represents the agreement of a metric compared to the human judgements.

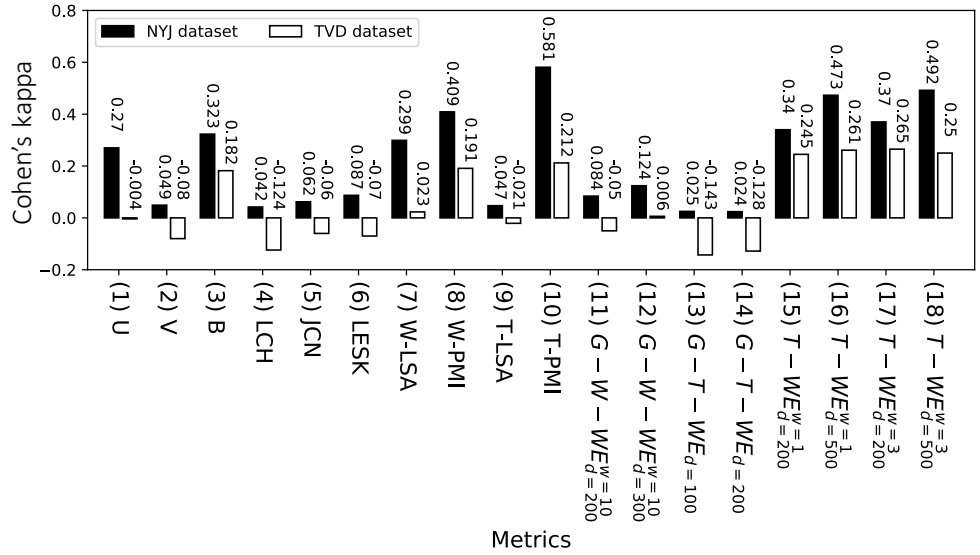

Figure 4.6: The Cohen's $kappa$ agreement between the human judgements and the 18 metrics. Each bar in the figure represents a Cohen's $kappa$ score of a metric compared to the human judgements.

across the two datasets. For instance, our T-WE$^{w=3}_{d=500}$ metric performs second best in the NYJ dataset (see Metric (18) in Figures 4.5 and 4.6). In the TVD dataset, all of our T-WE metrics outperform the rest of metrics while the performance of T-PMI (the best performing metric on the NYJ dataset) is rather limited. As mentioned in Section 4.5.1, the TVD dataset has more noise compared to the NYJ dataset and therefore the topics from the TVD

dataset are more difficult for human assessors to distinguish. However, our T-WE metrics (Metrics (15)-(18) in Figures 4.5 and 4.6)) can still have a higher agreement and Cohen's $kappa$ scores compared to all the rest of metrics. This indicates the high effectiveness of our T-WE metrics. There can be two reasons for such results. First, our Twitter background dataset (c.f. Section 4.5.1) covers more background knowledge[20] than the Wikipedia pages and the other used Twitter background dataset, the $GloVe$ Twitter dataset. This can be seen in the performance differences between Metrics (11)-(14) (using the Wikipedia pages and the $GloVe$ Twitter dataset as their external resources) and our proposed T-WE metrics, i.e. Metrics (15)-(18) in Figures 4.5 and 4.6. This answers our second research question (**RQ2** in Section 4.5.3), i.e. our Twitter background dataset helps our T-WE metrics to better identify the topical coherence compared to the metrics using the other resources. Second, using word embeddings is more effective than a LSA model when computing the word similarities. This indicates the higher effectiveness of the WE-based metrics (i.e. T-WE and W-WE) compared to the LSA-based metrics (i.e. T-LSA and W-LSA). We also observe that, in our T-WE metrics, a metric with a higher size of context window (w) and a higher dimension size of word embedding vectors (d) have slightly higher agreement and Cohen's $kappa$ scores[21]. For example, the agreement and Cohen's $kappa$ scores of T-WE$^{w=3}_{d=500}$ are better than those of T-WE$^{w=3}_{d=200}$ on the NYJ dataset. This answers our third research question (**RQ3** in Section 4.5.3), i.e. a metric using a WE model trained with a higher w and a higher d better aligns with the human judgements.

Note that apart from the WE-based metrics, T-PMI performs second best. The statistical metrics (U, V and B) and the metrics using WordNet (LCH, JCN and LESK) cannot accurately determine topic preferences like human assessors since their agreement and Cohen's $kappa$ scores are rather low across the two used datasets, compared to our proposed coherence metrics (Metrics (9)-(18) in Table 4.6).

4.5.4.3 Ranking Comparison of the Topic Modelling Approaches

As discussed in Section 4.4.4, we now report how these metrics can distinguish the performance differences between the three used topic modelling approaches. For each *comparison unit*, we generate 100 topic pairs per dataset (see Table 4.5). Tables 4.7 and 4.8 report the average coherence scores of the 100 topics from the three types of topic modelling approaches calculated using the 18 automatic coherence metrics (see Table 4.6) in the NYJ and TVD datasets, respectively. We also average the fraction of human votes for each of the three

[20]The period of our Twitter background dataset covers the period of our two Twitter datasets.

[21]Except that T-WE$^{w=3}_{d=500}$ metric is worse than T-WE$^{w=1}_{d=500}$ metric in terms of Cohen's $kappa$f as shown in Figure 4.6.

Table 4.7: The results of the automatic topic coherence metrics on the **NYJ** dataset and the corresponding ranking orders. The values in the column of a metric are the coherence scores calculated by this metric. "×" means no statistically significant differences ($p \leq 0.05$, Wilcoxon signed-rank test, see Section 4.4.4) among the three topic modelling approaches. Two topic modelling approaches have the same rank if there are no significant differences between them. A metric is in bold if the ranking order of this metric matches/partly matches the order from the human assessors.

	(1) U	Rank	(2) V	Rank	(3) B	Rank	Humans	Rank
LDA	0.092		0.548		1.365	1^{st}	0.636	1^{st}
TLDA	0.196	×	0.529	×	0.828	2^{nd}	0.553	2^{nd}
PAM	-0.074		0.542		-3.473	3^{rd}	0.129	3^{rd}

	(4) LCH	Rank	(5) JCN	Rank	(6) LESK	Rank
LDA	0.517		0.020		0.028	
TLDA	0.494	×	0.019	×	0.018	×
PAM	0.544		0.021		0.009	

	(7) **W-LSA**	Rank	(8) W-PMI	Rank	(9) T-LSA	Rank	(10) **T-PMI**	Rank
LDA	0.157	$1^{st}/2^{nd}$	0.205	1^{st}	0.014		1.63e-3	1^{st}
TLDA	0.132	$1^{st}/2^{nd}$	0.190	2^{nd}	0.004	×	1.52e-3	2^{nd}
PAM	0.073	3^{rd}	0.150	3^{rd}	0.011		4.53e-4	3^{rd}

	(11) G-W-WE$^{w=10}_{d=200}$	Rank	(12) **G-W-WE**$^{w=10}_{d=300}$	Rank	(13) G-T-WE$_{d=100}$	Rank	(14) G-T-WE$_{d=200}$	Rank
LDA	0.168		0.129	1^{st}	0.266		0.240	
TLDA	0.157	×	0.126	$2^{nd}/3^{rd}$	0.252	×	0.225	×
PAM	0.160		0.117	$2^{nd}/3^{rd}$	0.259		0.234	

	(15) **T-WE**$^{w=1}_{d=200}$	Rank	(16) **T-WE**$^{w=1}_{d=500}$	Rank	(17) **T-WE**$^{w=3}_{d=200}$	Rank	(18) **T-WE**$^{w=3}_{d=500}$	Rank
LDA	0.088	$1^{st}/2^{nd}$	0.068	$1^{st}/2^{nd}$	0.085	$1^{st}/2^{nd}$	0.065	$1^{st}/2^{nd}$
TLDA	0.100	$1^{st}/2^{nd}$	0.080	$1^{st}/2^{nd}$	0.098	$1^{st}/2^{nd}$	0.080	$1^{st}/2^{nd}$
PAM	0.074	3^{rd}	0.053	3^{rd}	0.071	3^{rd}	0.050	3^{rd}

topic modelling approaches, shown in Tables 4.7 and 4.8 as column "Humans" (shown in grey background). We apply the methodology introduced in Section 4.4.4 to obtain the ranking orders shown as column "Rank" in Tables 4.7 and 4.8.

By comparing the human ground-truth ranking orders of the three topic modelling approaches, we observe that the three topic modelling approaches perform differently over the two datasets. The human ground-truth ranking order is LDA^{1st}>TLDA^{2nd}>PAM^{3rd} and TLDA^{1st}>LDA$^{2nd/3rd}$>PAM$^{2nd/3rd}$ for the NYJ and TVD datasets, respectively.

First, the ranking order by the PMI-based (T-PMI) metric using our Twitter background dataset matches the ground-truth ranking order across the two datasets. In addition, our proposed WE-based metric (T-WE) almost performs the same as T-PMI except that the corresponding ranking order only partly matches the ground-truth ranking order for the NYJ dataset (see metric numbered (18) vs. "Humans" in Table 4.7). This suggests that both the T-PMI and our T-WE metrics can distinguish the performance differences of the three topic modelling approaches like the human assessors. Second, the LSA-based metric using Wikipedia pages as an external resource (W-LSA) can also match or partly match the ground-truth ranking order. This suggests that W-LSA has the capacity to assess the coher-

Table 4.8: The results of the automatic topic coherence metrics on the **TVD** dataset and the corresponding ranking orders. The values in the column of a metric are the coherence scores calculated by this metric. "×" means no statistically significant differences ($p \leq 0.05$, Wilcoxon signed-rank test, see Section 4.4.4) among the three topic modelling approaches. Two topic modelling approaches have the same rank if there are no significant differences between them. A metric is in bold if the ranking order of this metric matches/partly matches the order from the human assessors.

	(1) U	Rank	(2) V	Rank	(3) B	Rank	Humans	Rank
LDA	0.293	$1^{st}/2^{nd}$	0.548		-1.31	$1^{st}/2^{nd}$	0.475	$2^{nd}/3^{rd}$
TLDA	0.248	3^{rd}	0.535	×	-0.606	$1^{st}/2^{nd}$	0.590	1^{st}
PAM	0.304	$1^{st}/2^{nd}$	0.515		-2.092	3^{rd}	0.431	$2^{nd}/3^{rd}$

	(4) LCH	Rank	(5) JCN	Rank	(6)LESK	Rank
LDA	0.448		0.017		0.014	
TLDA	0.434	×	0.016	×	0.014	×
PAM	0.502		0.020		0.016	

	(7) **W-LSA**	Rank	(8) W-PMI	Rank	(9) T-LSA	Rank	(10) **T-PMI**	Rank
LDA	-0.019	$2^{nd}/3^{rd}$	0.134	$1^{st}/2^{nd}$	-0.033		3.57e-4	$2^{nd}3^{rd}$
TLDA	0.064	1^{st}	0.141	$1^{st}/2^{nd}$	-0.019	×	4.11e-4	1^{st}
PAM	-0.041	$2^{nd}/3^{rd}$	0.127	3^{rd}	-0.023		3.26e-4	$2^{nd}/3^{rd}$

	(11) G-W-WE$_{d=200}$	Rank	(12) G-W-WE$_{d=300}$	Rank	(13) G-T-WE$_{d=100}$	Rank	(14) G-T-WE$_{d=200}$	Rank
LDA	0.126		0.094		0.222		0.200	
TLDA	0.113	×	0.086	×	0.211	×	0.190	×
PAM	0.127		0.094		0.225		0.203	

	(15) **T-WE**$^{w=1}_{d=200}$	Rank	(16) **T-WE**$^{w=1}_{d=500}$	Rank	(17) **T-WE**$^{w=3}_{d=200}$	Rank	(18) **T-WE**$^{w=3}_{d=500}$	Rank
LDA	0.086	$2^{nd}/3^{rd}$	0.064	$2^{nd}/3^{rd}$	0.076	$2^{nd}/3^{rd}$	0.058	$2^{nd}/3^{rd}$
TLDA	0.094	1^{st}	0.071	1^{nd}	0.082	1^{st}	0.064	1^{st}
PAM	0.080	$2^{nd}/3^{rd}$	0.063	$2^{nd}/3^{rd}$	0.076	$2^{nd}/3^{rd}$	0.057	$2^{nd}/3^{rd}$

ence of topics. Note that the ranking orders from all the other metrics fail to match or partly match the ground-truth ranking order.

4.5.4.4 Summary

In Sections 4.5.4.2 and 4.5.4.3, we have evaluated the performance of 18 coherence metrics (i.e. the 18 metric variants listed in Table 4.6) by identifying whether they have a high agreement with humans on topic preferences and whether they can rank the performance of the three topic modelling approaches in a manner that is aligned with human assessors in terms of topical coherence. For the topic preference agreement, our proposed T-WE metrics (e.g. T-WE$^{w=3}_{d=500}$ using our Twitter background dataset as an external resource) perform best on the TVD dataset and second best on the NYJ dataset. Our proposed T-PMI preforms best on the NYJ dataset, however, its performance for a noisier dataset (TVD) is rather low (see the white histograms in Figures 4.5 and 4.6). This suggests that T-PMI does not appear to work consistently across two different datasets. In terms of the ranking order of the three topic modelling approaches, T-PMI can obtain the same ranking order as the ranking order

from the human assessors, while our T-WE metrics (Metrics (15)-(18) in Tables 4.7 and 4.8) perform second best and their performance are very close to T-PMI.

In practice, we want to use a metric, which can consistently deal with different types of datasets. Although T-PMI can accurately rank the performance differences of the three topic modelling approaches, its performance is rather limited on the TVD dataset (i.e. the dataset with more noise) when directly identifying the more coherent topics from the topic pairs. On the other hand, our T-WE metric (T-WE$^{w=3}_{d=500}$) performs the best in terms of the coherence agreement scores (c.f. Section 4.5.4.2). It also performs second best when identifying the differences of the three topic modelling approaches across the two used datasets (c.f. Section 4.5.4.3). Indeed, our T-WE metric works equally effectively across the two used datasets. Hence, in answering the first research question (i.e. **RQ1** in Section 4.5.3), we conclude that our T-WE metric (T-WE$^{w=3}_{d=500}$) is consistently effective across the two used Twitter datasets, while being highly aligned with the human assessors.

It is also worth mentioning that the PMI-based metrics leverage the PMI data of a few hundred millions of word pairs (e.g. T-PMI and W-PMI, see Table 4.6). It can be difficult to store[22] and use. On the other hand, our T-WE metrics use a few hundred thousand (the size of the vocabulary) word vectors in the WE models, which means that the WE model can be much easier to store and use. Moreover, a WE model is faster to train than a LSA model (c.f. Section 4.3.2). In the rest of this thesis, we use our proposed T-WE metrics to evaluate the coherence of topics generated from Twitter. In particular, we use the T-WE$^{w=3}_{d=500}$ metric due to its high effectiveness. For simplicity, we denote the T-WE$^{w=3}_{d=500}$ metric as the word embedding-based metric or the WE-based metric in the following chapters.

4.6 Evaluating the Global Coherence of Topic Models

In the previous sections, we have addressed how to evaluate the coherence of a single topic in a topic model, where a topic model is generated using a topic modelling approach and contains K topics. In the literature, previous work has evaluated the global coherence of a topic model by averaging the coherence scores of all of the K topics (e.g. in Yan et al., 2013; Cheng et al., 2014; Sridhar, 2015). However, little work has studied the most coherent topics in a topic model when assessing the coherence of a topic model as a whole. When social scientists apply topic modelling to extract topics from Twitter, they expect to examine the most coherent topics from a generated topic model rather than all the topics. Therefore, we propose a `coherence at n` metric to evaluate the global coherence of a topic model by assessing the top n ranked topics.

[22] It takes a lot of space to store.

4.6.1 `Coherence at n` Metric

To evaluate the coherence of a topic model, the average coherence score of the K topics in a topic model can be used intuitively to evaluate a topic model (e.g. in Yan et al., 2013). However, in practice, there can be many topics in a topic model. When a user wants to examine the topics in a topic model, they can be only interested in the most coherent topics as the incoherent topics can waste users' time. Although the average coherence of a topic model reflects the quality of the entire topic model, it cannot effectively indicate the coherence quality of a topic model from the user perspective, i.e. users are interested in the top ranked topics by their coherence. We argue that it is more effective to evaluate the coherence of topic models by computing the average coherence of the top-ranked topics. Inspired by ranking metrics such as the `precision at k` metric (Manning et al., 2008a), we use `coherence at n` to evaluate the coherence of a topic model. In particular, `coherence at n` (coherence@n) indicates the average coherence score of the top n most coherent topics, where all topics are ranked by their coherence scores. We argue that `coherence at n` can more effectively capture the coherence of a Twitter topic model. In the following sections, we evaluate our `coherence at n` metric through a user study.

4.6.2 Experiments

In this section, we first describe the used datasets and experimental setup, followed by the results of the experiments.

4.6.2.1 Datasets

We use two datasets in our experiments. The first one is comprised of the tweets of 2,452 newspaper journalists in New York posted from 01/05/2015 to 31/05/2015, denoted here as MAY. This dataset is a smaller version of the NYJ dataset in Section 4.5.1. We reduce the size of the dataset in order to decrease the computation time of running a topic modelling approach. Since the MAY dataset has still a large number of tweets discussing sufficient topics, reducing the size of the dataset might not have a significant impact on the result[23]. We also use the same TVD dataset from Section 4.5.1. The details of these two datasets are shown in Table 4.9.

[23]We also apply the topic modelling approach on a Twitter dataset with a lot larger size in Chapter 7. We obtain a similar conclusion as this section.

Table 4.9: Two used Twitter datasets for examining the top-ranked topics.

Name	Time Period	The number of users	The number of tweets
MAY	01/05/2015 - 31/05/2015	2,452	334,922
TVD	8pm-10pm 02/04/2015	121,594	343,511

4.6.3 Experimental Setup

In this section, we explain how we generate topics using topic modelling approaches (Section 4.6.3.1) and the used Twitter topic coherence metrics (Section 4.6.3.2). We list our research questions in Section 4.6.3.3.

4.6.3.1 Generating Topics

We apply LDA and TLDA as in Section 4.5. However, we do not apply PAM as its performance is rather low (see Tables 4.7 and 4.8). In addition, we apply the pooling strategy (see Section 3.2.2.2) on the LDA approach (denoted as PLDA), since the pooling strategy has been reported to improve the coherence of topics (see Section 3.2.2). For PLDA, we group the tweets posted by the same user in a given time interval into a virtual document[24]. The time interval is set to 10 minutes for TVD, and 6 hours for MAY, given the narrow time (two hours) period of the TVD dataset and the more expansive one (one month) for the MAY dataset. We set the LDA parameters α and β and the TLDA parameter γ as in Section 4.5.2. We vary the number of topics K in our experiments, which allows us to examine how the coherence of a topic model changes when K changes. Since the TVD dataset contains just two hours of tweets, we set the maximum topic number K to 100, and then use 46 different K values between 10 and 100 (step = 2). We set K to a maximum of 500 for MAY, and use 49 different K values ranging from 10 to 500 (step = 10). Each topic modelling approach is run 5 times for each K. Thus, we obtain 5 topic models for each K. In the next section, we analyse the coherence of these 1,425 topic models ($46 \times 5 \times 3 + 49 \times 5 \times 3 = 1,425$).

4.6.3.2 Coherence Metrics Setup

We apply the best-proposed word embedding (WE)-based metric (i.e. T-WE$^{w=3}_{d=500}$) in this experiment as it has a high-level agreement with the human coherence judgements as shown in Section 4.5. When evaluating the coherence of a topic model, the WE-based metric is applied first to calculate the coherence scores of all topics and then the `coherence at n`

[24]We do not group tweets by hashtags since there are not many hashtags in our datasets.

(coherence@n) metric averages the n most coherent topics, where n is set to $\{5, 10, 20, 30, 40\}$ in our experiments. As a baseline, we use the average coherence.

4.6.3.3 Research Questions

We aim to answer two research questions:

- **RQ4.** Does a topic model with a higher K generate topics with a higher coherence than when a smaller K is used?
- **RQ5.** Is our coherence@n metric more effective than the average coherence metric?

4.6.4 Analysis of the Top Ranked Topics

Figure 4.7 shows the average coherence and `coherence at n` scores of topic models generated from the two used Twitter datasets using the three applied topic modelling approaches (LDA, TLDA and PLDA). First, on analysing Figure 4.7, it is clear that the average coherence (the solid black line) of all topics in a model does not change much as K increases across the three topic modelling approaches. These results are similar to the observation of Stevens et al. (2012). However, the coherence@n scores (represented by coloured lines with distinguishing symbols) of all topic models increases as K grows. This suggests that the topic modelling approaches generate topics with a higher coherence when K increases. However, if the average coherence metric is used when evaluating the coherence of topic models, users might not notice that a topic model with a higher K contains topics with a higher coherence and therefore might choose a topic model with a smaller K. This is because the average coherence metric suggests that a topic model with a smaller K is better. For example, the black line (the average coherence) in Figure 4.7 (b) decreases when K increases, which indicates that a topic model with a smaller K has a higher coherence. In fact, when a higher number of topics K is set, the coherence of the n most coherent topics increases. However, the average coherence cannot effectively indicate the difference of these topic models. Here, coherence@n is preferred. Second, the coherence@n score is higher for TLDA across the two datasets, which suggests that the top n topics in the TLDA models have a higher coherence. We also see that PLDA generates more coherent topics than LDA. Indeed, PLDA performs similarly to TLDA, particularly on the TVD dataset. For example, the patterns in Figure 4.8 (b) and (c) are similar. Third, we observe that the average coherence and coherence@n scores of the LDA & PLDA topic models on the TVD dataset become stable around K=80, while the coherence of the TLDA topic models on the MAY dataset has a local peak around K=440. A larger K (e.g. $K > 80$ in TVD or $K > 440$ in MAY) seems not to help to generate a topic model with a significant higher coherence. Since

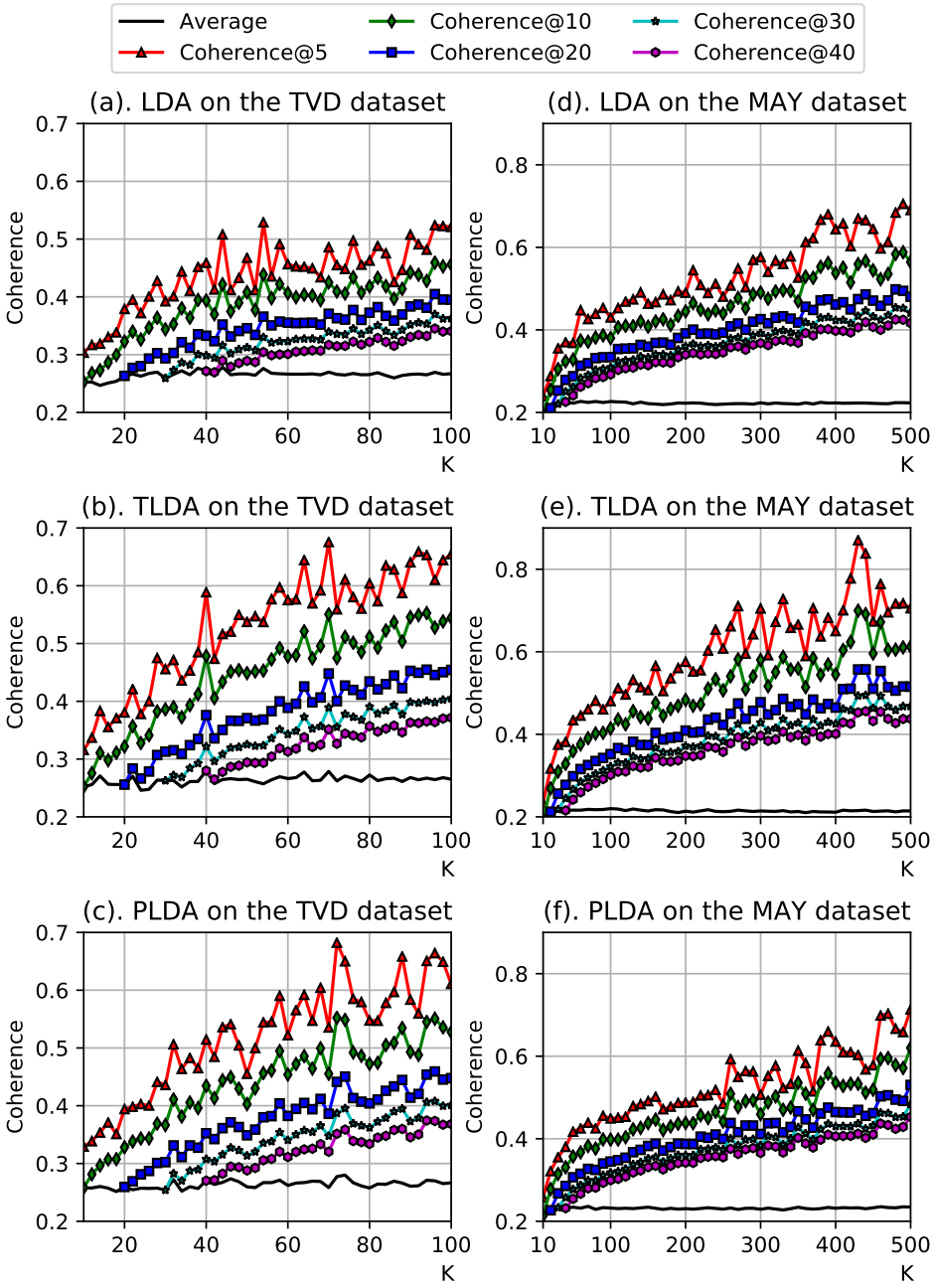

Figure 4.7: The coherence of three types of topic models with different values of K over two datasets.

a larger K leads to a higher computational cost, the K value should be selected when the coherence of the topic model begins to stabilise or when it reaches a local peak.

Next, in Figure 4.8, we show the distributions of the topics' coherence scores for LDA, TLDA and PLDA topic models with varying K on the TVD dataset[25]. The coherence scores

[25]We observe a similar result on the MAY dataset. Hence, we do not list the figure of MAY.

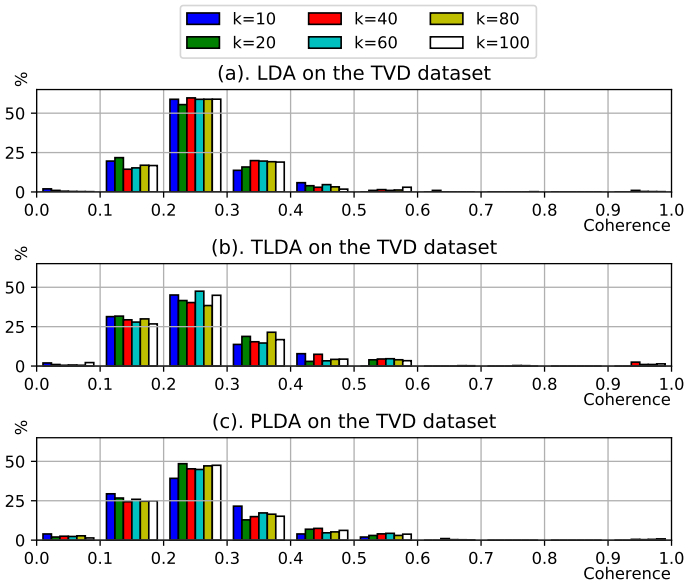

Figure 4.8: The coherence value distributions in the study of examining the top-ranked topics.

are distributed across 10 bins, to show the proportion of topics exhibiting different levels of coherence. First, the volume of topics with coherence $[0.1, 0.3)$ is highest across all topic models on the TVD dataset. This suggests that the majority of topics generated from our Twitter data (TVD) can be incoherent. This motivates us to investigate a tailored topic modelling approach for Twitter in the next chapter. As K increases, the topic models include more topics with less coherence[26]. This is why the average coherence of the topic models decreases as K increases, shown in Figure 4.7 by the solid black lines. Second, when K increases, the topic modelling approaches indeed can generate more topics with a higher coherence. This is indicated by a high number of topics in the coherence bins [0.4, 0.6)[26].

To summarise, we observe that the generated topic models with a higher K contain topics with a higher coherence according to our coherence@n metric. On the other hand, the average coherence metric conveys the opposite. To verify that a larger K indeed helps to generate topics that are easier for users to interpret, in the next section, we conduct a user study where we ask human assessors to choose the topics with a higher coherence from the given topic pairs.

[26]Note that the y-axis in Figure 4.8 indicates the percentages of topics in a coherence bin.

4.6.5 User Study

We conduct a pairwise preference user study, similar to the user study in Section 4.4.2. We recruited workers from CrowdFlower[27] and asked them to select the most coherent topic from two provided topics (a topic pair). The setting of this user study is the same as the user study in Section 4.4.2. Next, we describe how we generate the topic pairs in Section 4.6.5.1 and report the results of user study in Section 4.6.5.2.

4.6.5.1 Generating Topic Pairs

We use the same way for generating topic pairs as in Section 4.4.1. The difference is that we generate topic pairs of the three topic modelling approaches in Section 4.4.1 while we generate topic pairs of TLDA models with different K generated from the MAY dataset in this user study. We select the TLDA topic models since TLDA was shown to generate topics with the highest coherence scores (see Figure 4.7 (e)). In Section 4.5, we show that the MAY dataset is less noisy than the TVD dataset and that the topics from the MAY dataset are easier for human assessors to interpret. Hence, to ease the tasks of the workers in the crowdsourced user study, we select the TLDA models generated from the MAY dataset.

We compare topic models with K=a (recall that each approach is repeated 5 times per K, see Section 4.6.3.1) and topic models with K=b, i.e. comparison Unit(a,b) (a<b). Therefore, we can examine whether topic models with higher K (b) have more coherent topics than topic models with a smaller K (a). From each selected topic model, we select the top n most coherent topics using the WE-based metric. Thus, we have two topic pools: $P_{K=a}$ and $P_{K=b}$, where each pool has $5 \times n$ topics. We use the same method as in Section 4.4.1 to generate a number of topic pairs for Unit(a,b), i.e. Pairs($P_{K=a} \rightarrow P_{K=b}$) and Pairs($P_{K=b} \rightarrow P_{K=a}$) from $P_{K=a}$ and $P_{K=b}$. In our user study, we compare the coherence of topic models with K=50 vs. K=300 (Unit(50,300)) and topic models with K=100 vs. K=390 (Unit(100,390)). To generate the topic pairs for Unit(50,300), we select top 30 (i.e. n=30) topics of topic models with K=50 and K=300 to generate 40 topic pairs, i.e. 20 Pairs($P_{K=50} \rightarrow P_{K=300}$) and 20 Pairs($P_{K=300} \rightarrow P_{K=50}$), from $P_{K=50}$ and $P_{K=300}$ (each pool has 5 $\times$ n, i.e. 150 topics). On the other hand, we choose the top 20 topics of topic models with K=100 and K=390 for Unit(100,390)[28] and generate another 40 topic pairs from $P_{K=100}$ and $P_{K=390}$ (each pool has 100 topics). Finally, we generate 40 topic pairs for each comparison unit. In this user study, we use the same user interface and instructions as in Section 4.4.2. Figure 4.9 shows an example of a topic pair.

[27] `https://www.crowdflower.com`

[28] The top 20 topics in Unit(100,390) are more distinguishable than the top 30.

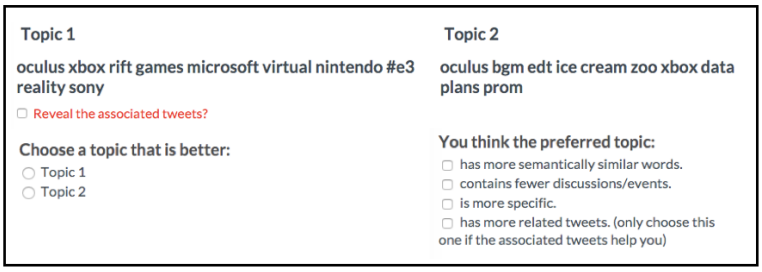

Figure 4.9: The CrowdFlower user interface for studying the top-ranked topics.

Table 4.10: The comparison of coherence scores given by our coherence@n metric and human assessors. $*/(**)$ denote $p < 0.01/(p < 0.05)$ according to the Wilcoxon signed-rank test, compared to the smaller K.

(a). TLDA topic models with K=50 vs. K=300, Unit(50,300)

K	Human vote fraction	Coherence@30	Average Coherence
50	0.311	0.266	0.233
300	$\mathbf{0.689}^{(*)}$	$\mathbf{0.428}^{(*)}$	0.225

(b). TLDA topic models with K=100 vs. K=390, Unit(100,390)

K	Human vote fraction	Coherence@20	Average Coherence
100	0.411	0.317	0.231
390	$\mathbf{0.589}^{(**)}$	$\mathbf{0.436}^{(**)}$	0.223

4.6.5.2 Crowdsourcing Results

Table 4.10 shows the human judgement results compared with the coherence@n scores using the WE-based metric. In total, we obtain 801 judgements from 52 different trusted workers. For comparison unit (50,300) - Table 4.10 (a) - the 40 topics we select the from topic models with K=300 are significantly more coherent than those from the topic models with K=50 according to both the human vote[29] fraction and the coherence@30 scores. We observe the same results for comparison unit (100,390) (see Table 4.10 (b)). Both the human assessors and our coherence@n metric suggest that the topic models with a larger K have more coherent topics (i.e. topic models with $K = 300$ and $K = 390$) than topic models with a smaller K (i.e. topic models with $K = 50$ and $K = 100$), which answers the fourth research question (**RQ4** in Section 4.6.3.3), i.e. a topic model with a higher K generates more coherent topics than a topic model with a smaller K. However, the average coherence metric (listed in Table 4.10 as column "Average Coherence") conveys the opposite, which does not align with the human assessors. Hence, we answer the fifth research question (i.e. **RQ5** in Section 4.6.3.3) and conclude that our coherence@n metric is more effective than the average coherence metric in that it is more aligned with human judgements.

[29]A topic in a topic pair receives one vote when it is preferred by a human assessor.

4.7 Conclusions

In this chapter, we have investigated approaches to evaluate the coherence of topics from tweets and the coherence of a topic model containing K topics. To more effectively evaluate the coherence of topics, we proposed two methods to improve the semantic similarity-based topic coherence metrics: 1) using a Twitter background dataset as an external resource and 2) using word embeddings (WE). Based on these two methods, we proposed 4 topic coherence metrics (including 10 variants) for tweet topics (see Table 4.6). To evaluate the performance of the 4 proposed topic coherence metrics compared to 8 existing metrics, we conducted a large-scale pairwise user study to obtain the human judgements. We identified that the WE-based metric using a Twitter background dataset works consistently across the two used Twitter datasets when evaluating the topical coherence and the differences of the three topic modelling approaches (c.f. Figures 4.5 & 4.6 and Tables 4.7 & 4.8). We concluded that the WE-based metric is effective when assessing the coherence of topics generated from Twitter. On the other hand, in order to evaluate the coherence of a topic model containing K topics, we proposed a `coherence at n` metric to assess the global coherence of a topic model by averaging the top-ranked topics in a topic model. By conducting a large-scale experiment on two Twitter datasets, we showed that the `coherence at n` metric can more effectively evaluate the coherence of a topic model compared to the more commonly used average coherence score (c.f. Figure 4.7 and Table 4.10). We recommend to use our `coherence at n` metric when evaluating the global coherence of topic models generated from tweets.

In the next chapter, we aim to develop a tailored topic modelling approach for Twitter data, which can generate topics with a higher coherence. This topic modelling approach identifies the 'what' and addresses the 'when', i.e. the time dimension of tweets. The proposed WE-based metric (i.e. T-WE$^{w=3}_{d=500}$ in Table 4.6) in this chapter will be used to evaluate the coherence of the topic modelling approach proposed in the next chapter.

Chapter 5

Time-Sensitive Topic Modelling

5.1 Introduction

In this chapter, we investigate a topic modelling approach for Twitter data, which addresses the 'what' and 'when' in our thesis statement. As discussed in Section 3.2.2, it is challenging to generate coherent topics from Twitter. Scholars wrestled with the appropriate tools for best capturing the topics of discussion conveyed on Twitter (e.g. in Hong and Davison, 2010; Zhao et al., 2011b; Sokolova et al., 2016). We propose a time-sensitive topic modelling approach, which can effectively generate topics with a high coherence from Twitter data and that can be easy to interpret by humans.

Topics on Twitter can be discussed in different time periods and therefore their popularity differs over time. During the topic modelling generative process, instead of only assigning words to a tweet, we can also assign the timestamps of the words occurring in the tweet to indicate the usage and popularity of these words during an interval of time. This is because a tweet is associated with a timestamp on Twitter when a user posts the tweet. The integration of the time dimension of tweets allows a topic modelling approach to be sensitive to time, i.e. modelling topics by considering both the usage of words and when precisely the topic is popular, thus topics can be distinguished during the event thereby becoming more interpretable by end-users. Hence, in this thesis, we argue that the use of the time dimension of tweets (i.e. addressing the 'when') during the topic modelling process allows to generate topics that can be easy for humans to interpret. In particular, we first study the Variational Bayesian (VB, introduced in Section 2.2.3) implementation approach of topic modelling. Building on the VB implementation, we then propose our time-sensitive VB implementation

approach of topic modelling for Twitter data (called TVB[1]), which embraces the time dimension of tweets. We extend the classical VB approach by incorporating the Beta distribution to model the time dimension of tweets. To balance the weights of the words and the time of tweets, we propose a *balance* parameter to control their impacts during the topic modelling process. We evaluate our TVB approach compared to 4 existing widely used topic modelling baseline approaches, such as Twitter LDA (introduced in Section 3.2.2.1) and Topics Over Time (introduced in Section 3.2.2.3). We conduct experiments on two real-world Twitter datasets and evaluate both the baselines and our proposed TVB approach in terms of topical coherence and the extent to which the generated topics are mixed. We show that our TVB approach is overall promising and effective when generating coherent and human interpretable topics from Twitter. The rest of this chapter is organised as follows:

- Section 5.2 introduces our time-sensitive topic modelling approach.
- Section 5.3 positions our time-sensitive approach with respect to existing time-sensitive and Twitter topic modelling approaches.
- Section 5.4 introduces our experiments to evaluate both our time-sensitive topic modelling approach and the baseline approaches.
- Section 5.5 reports and analyses the results of our experiments.
- Section 5.6 summarises the conclusions of this chapter.

5.2 Integrating the Time Dimension of Tweets

In this section, we introduce our **t**ime-sensitive topic modelling approach implemented using the **V**ariational **B**ayesian implementation approach. We denote our approach as TVB. Our approach extends the classical VB approach by integrating the time dimension of tweets. The time dimension of tweets helps to capture the topical trend information. We first give the definition of a topical trend (Section 5.2.1). Then, we explain how we model the topical trends (Section 5.2.2) and how we integrate time (Section 5.2.3) in our TVB approach.

[1]In this chapter, we simply use TVB to denote our proposed time-sensitive topic modelling approach implemented by Variational Bayesian. We use VB to denote the classical topic modelling approach implemented by Variational Bayesian.

5.2.1 Topical Trends

A topic can contain a collection of tweets and each tweet is associated with a timestamp indicating the created/posted time of the tweet. Therefore, each topic is associated with a collection of timestamps, which can be used to estimate the popularity of this topic. The popularity of a topic can then be measured as the frequency of tweets within this topic during the time intervals, denoted by time series $pop^N = \{\langle itv_1, c_1\rangle, ..., \langle itv_n, c_n\rangle, ..., \langle itv_N, c_N\rangle\}$, where c_n is the number of tweets posted during time interval itv_n. This representation of time series is also used in Yang and Leskovec (2011); Ma et al. (2013); Kong et al. (2014), where they used time series to analyse or predict the topical trends. The popularity of a topic indicates the trend information of this topic during a time period. Therefore, we call it the *topical trend*. We model the topical trend as a continuous probability distribution, which indicates how likely a topic can happen during a time interval. Theoretically, any continuous distribution can be used to simulate the topic proportion over time. However, to better estimate topical trends, the continuous distribution has to approximate the real topical trends. Indeed, recently, the Beta distribution has drawn a lot of attention for accommodating a variety of shapes given an x-axis interval (Guolo et al., 2014). Therefore, we choose to use the Beta distribution since it can more accurately fit the various shapes of topical trends[2]. The topical trend of a topic k can be represented by τ parametrised by two shape parameters ρ_k^1 and ρ_k^2 in Equation (5.1):

$$\tau_k = Beta(\rho_k^1, \rho_k^2) = \frac{1}{B(\rho_k^1, \rho_k^2)} t^{\rho_k^1 - 1}(1-t)^{\rho_k^2 - 1} \tag{5.1}$$

where $Beta$ means the beta distribution, B is the Beta function and t is the timestamp of a tweet. In this thesis, we use "*topical trend*" or "*trend*" to generally indicate the topic popularity over time. The topical trend estimated by the Beta distribution is called the *estimated trend* of a topic while the real popularity of a topic over time is called the *real trend* of a topic. Figure 5.1 shows an example of a real trend and its estimated trend of a topic[3]. The topic "InternationalWomensDay" is associated with tweets containing the keyword "International-WomensDay" posted from 08/03/2017 07:39 to 09/03/2017 07:39. The histograms show the real topical trend while the solid line is the estimated trend drawn by the beta distribution.

[2]We also use other distributions to fit the time trend, such as the normal and gamma distributions. We found that they do not perform better than the Beta distribution when fitting the time trend. A real topic can have a complicated popularity curve, e.g. multiple peaks and valleys. In this case, the Beta distribution might not effectively fit the curve. We acknowledge this limitation of the Beta distribution in TVB. We do not use the non-parametric statistical distribution in order to simplify the inference of our TVB approach.

[3]The histograms indicate the normalised density information instead of the number of tweets.

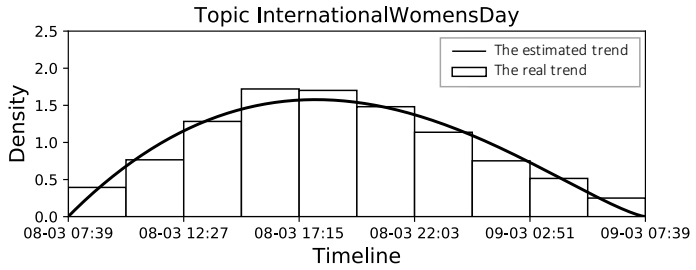

Figure 5.1: An example of topical trend.

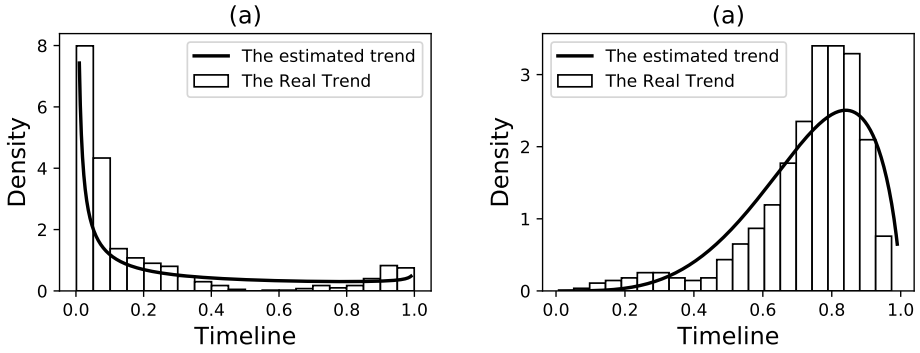

Figure 5.2: Examples of two topics with two different trends.

Topics can be discussed in different time periods on Twitter. For instance, Figure 5.2 shows two topic examples with different topical trends, where the timeline is normalised between 0 and 1. The real trends (indicated by the histograms in Figure 5.2) of these topics are different, i.e. the topic on the left of the figure was highly discussed at the very beginning of the timeline while the topic on the right of the figure has a peak at the end of the timeline. While we can naturally use the words of the topics for their identification, we can also use the topics' trends over time to distinguish between such topics. In this thesis, we model the trends of topics using the Beta distribution (i.e. the estimated trends indicated by the black lines in Figure 5.2) within our proposed TVB time-sensitive approach. As we will explain in Section 5.2, our TVB approach extracts latent topics by using not only the used words in the tweets but also their timestamps. The presence of time provides an additional feature to the topic modelling process allowing to group words within topics that occurred at a given time period. We further hypothesise that the use of time trends can help a topic modelling approach to generate topics with high coherence scores, since the topics can be better distinguished during a political event on Twitter.

Apart from increasing topical coherence, the integration of time into the topic modelling process allows to generate topics that are less *mixed*[4]. Figure 5.3 shows an example

[4]A mixed topic might contain several real topics about different issues/themes.

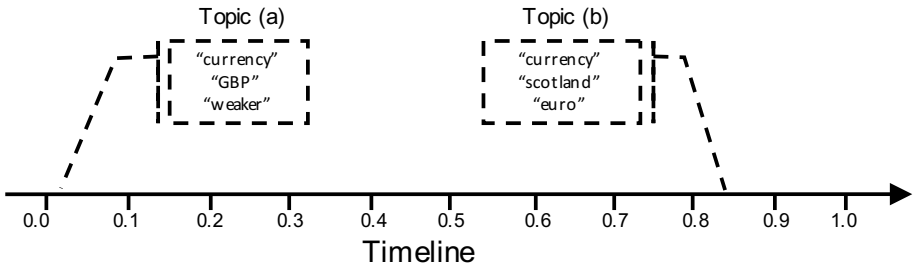

Figure 5.3: Examples of two topics in a timeline.

of two topics in the timeline (normalised between 0 and 1) during the Scottish Independence Referendum (Indyref) 2014 event. The two topics correspond to two popular discussion themes among Twitter users during the Indyref event. Topic (a) has a similar trend as the trend shown in Figure 5.2 (a). This topic was highly discussed at the beginning of IndyRef, as people were worried about the "*GBP*" "*currency*" becoming "*weaker*" because of Indyref (BBC, 2014). On the other hand, Topic (2) in Figure 5.3 was discussed at the end of the timeline (see the trend shown in Figure 5.2 (b)) where the Twitter users discussed the choice of the use of the "*euro*" as a "*currency*" in "*Scotland*" (Euractiv and Reuters, 2014). These two topics were discussed in different time periods and the involved users in these two topics are different, e.g. the Twitter users in Topic (b) are more likely to be Scottish. Indeed, a good topic modelling approach should identify these two topics as being different. However, since both topics are clearly related to "*currency*" and the words used in these two topics can be similar, they are likely to be mixed in a single topic if the time is not taken into account during the topic modelling process. For example, a mixed topic might look like: "*currency*", "*Scotland*", "*GBP*" and "*euro*". Although this mixed topic is coherent, it is more difficult for end-users to interpret with respect to the event compared to the separated topics (i.e. Topic (a) and (b)). One of the advantages of integrating time into topic modelling is that the aforementioned topics can be identified separately and distinguished. Hence, we argue that the use of the time dimension can help to alleviate the generation of mixed topics on Twitter data. In the next section, we introduce how we integrate the time dimension of tweets in our proposed TVB approach.

5.2.2 Our TVB Approach

To model the time trends of topics, we introduce the Beta distribution in TVB, where the Beta distribution indicates the possibility of a topic being discussed given a time period, i.e. the topic time distribution τ_k (k is the index of a topic while K is the total number of topics), parametrised by two shape parameters, $\rho_k = (\rho_k^1, \rho_k^2)$. The plate notation of our TVB

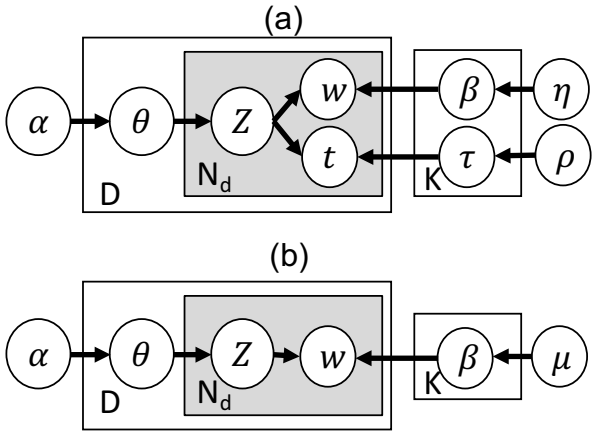

Figure 5.4: The plate notation of (a) the classical LDA and (b) our TVB approach.

approach is shown in Figure 5.4 (a), where $\boldsymbol{\eta}$ and $\boldsymbol{\alpha}$ are the hyperparameters for the topic term distributions $\boldsymbol{\beta}$[5] and the document topic distributions $\boldsymbol{\theta}$ (introduced in Section 2.2.1), respectively. In the classical LDA approach, only words are generated in the generative process using the topic term distributions and the document topic distributions (i.e. $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ in Figure 5.4 (b)). However, in our TVB approach, both words and their timestamps[6] are generated using $\boldsymbol{\beta}$, $\boldsymbol{\theta}$ and the topic time distributions $\boldsymbol{\tau}$. In particular, in our TVB approach, each word $w_{d,i}$ in the d-th document is assigned a topic assignment $z_{d,i}$ according to θ_d, where i is the word index. Since words ($\boldsymbol{w}$) in tweets are associated with timestamps ($\boldsymbol{t}$), a pair $(w_{d,i}, t_{d,i})$ is drawn from $\beta_{z_{d,i}}$ and $\tau_{z_{d,i}}$, respectively. We list all the used variables of TVB in Table 5.1 and we summarise the generative process of our TVB as follows:

1. Draw $\theta_d \sim$ Dirichlet(α), where $d \in \{1, ..., D\}$
2. Draw $\beta_k \sim$ Dirichlet(η), where $k \in \{1, ..., K\}$
3. Draw $\tau_k \sim (\rho_k^1, \rho_k^2)$, where $k \in \{1, ..., K\}$
3. For each word position d, i in a tweet, where $d \in \{1, ..., D\}$ and $i \in \{1, ..., N_d\}$:
   (a) Draw a topic assignment $z_{d,i} \sim \theta_d$
   (b) Draw a word $w_{d,i} \sim \beta_{z_{d,i}}$
   (c) Draw a timestamp $t_{d,i} \sim \tau_{z_{d,i}}$

This notion of integrating time is also applied in the literature, such as Topics over Time (ToT, see Section 3.2.2.3). The ToT approach uses the Beta distribution to integrate the time dimension in a sampling implementation approach. Our TVB approach also uses the Beta distribution, however, the implementation of our TVB approach is based on the

[5]A symbol in bold indicates all the topic terms distributions, e.g. $\boldsymbol{\beta} = \{\beta_1, ..., \beta_K\}$.
[6]The timestamp of a tweet is assigned to all the words in this tweet.

Table 5.1: The symbols used in our time-sensitive topic modelling approach.

Symbol	Description
K	The total number of topics.
k	The index of a topic out of K topics.
N	The size of vocabulary. n is the index of a word.
D	The total number of documents in a corpus. d is the index of a document.
N_d	The number of words in the d-th document $\vec{w_d}$.
i	The index of a term in a document or a corpus.
$w_{d,i}$	The i-th word in the d-th document.
$\vec{w_d}$	The d-th document.
$t_{d,i}$	The timestamp of the i-th word in the d-th document.
$\vec{t_d}$	The timestamps of the d-th document.
τ_k	The time distribution of topic k.
ρ_k^1/ρ_k^2	The hyperparameters of τ_k.
$z_{d,i}$	The topic assignment of $w_{d,i}$ and $t_{d,i}$. $\vec{z_d}$ are the assignments for $\vec{w_d}$.
θ_d	The topic distribution of the d-th document.
α_d	The hyperparameter of θ_d.
γ_d	The variational hyperparameter of θ_d.
β_k	The term distribution of topic k.
η_k	The hyperparameter of β_k.
λ_k	The variational hyperparameter of β_k.
$\phi_{d,i,k}$	The topic distribution of $w_{d,i}$.
δ	The *balance* parameter in TVB.

*The symbol is in bold when it indicates a collection of variables, e.g. $\boldsymbol{\beta} = \{\beta_1, ..., \beta_K\}$.

*The symbol without an index indicates a variable in general, e.g. β means a topic term distribution.

Variational Bayesian inference, which is different from the ToT approach. In the next section, we first describe the implementation of our TVB approach using Variational Bayesian implementation. Meanwhile, we explain the differences between our TVB approach and the related work in Section 5.3.

5.2.3 Implementation of Time-Sensitive Topic Modelling

Our TVB approach is based on the variational inference implementation approach. In a variational inference model, we have a variational topic term distribution $q(\beta_k|\lambda_k)$ (q is used to indicate the variational probability, introduced in Section 2.2.3), where λ_k is the variational hyperparameter of topic k. Similarly, there is a variational document topic distribution $q(\theta_d|\gamma_d)$ with γ_d as the variational hyperparameter. The core part of the variational inference approach is to minimise the distance between the two variational distributions ($q(\beta_k|\lambda_k)$ and

$q(\theta_d|\gamma_d)$) and their true distributions (the true topic term distributions $p(\beta_d|\eta_d)$ and the document topic distributions $p(\theta_d|\alpha_d)$) so that the variational distributions can be seen as the true distributions, i.e. minimising the distance between $q(\beta_k|\lambda_k)$ and $p(\beta_k|\eta_k)$ and minimising the distance between $q(\theta_d|\gamma_d)$ and $p(\theta_d|\alpha_d)$. In the next section, we describe how we minimise the distances.

5.2.3.1 Maximising the Lower Bound of a Document

We start with the log-likelihood of the d-th document, $log\ p(\vec{w}, \vec{t}|\boldsymbol{\alpha}, \boldsymbol{\eta}, \boldsymbol{\rho})$, shown in Equation (5.2)[7]:

$$\begin{aligned}
log\ p(\vec{w_d}, \vec{t_d}|\boldsymbol{\alpha}, \boldsymbol{\eta}, \boldsymbol{\rho}) &= log \int\int \sum_{z \in \vec{z_d}} p(\vec{w_d}, \vec{t_d}, \vec{z_d}, \theta_d, \boldsymbol{\beta}, \boldsymbol{\tau}|\alpha, \eta) d\theta_d d\boldsymbol{\beta} \\
&= log \int\int \sum_{z \in \vec{z_d}} p(\vec{w_d}, \vec{t_d}, \vec{z_d}, \theta_d, \boldsymbol{\beta}, \boldsymbol{\tau}|\alpha, \eta) q(\theta_d, \vec{z_d}, \boldsymbol{\beta}) / q(\theta_d, \vec{z_d}, \boldsymbol{\beta}) d\theta_d d\boldsymbol{\beta} \\
&\geq \int\int \sum_{z \in \vec{z_d}} q(\theta_d, \vec{z_d}, \boldsymbol{\beta}) log\ p(\vec{w_d}, \vec{t_d}, \vec{z_d}, \theta_d, \boldsymbol{\beta}, \boldsymbol{\tau}|\vec{\alpha}, \eta) d\theta_d d\boldsymbol{\beta} \\
&- \int\int \sum_{z \in \vec{z_d}} q(\theta_d, \vec{z_d}, \boldsymbol{\beta}) log q(\theta_d, \vec{z_d}, \boldsymbol{\beta}) d\theta_d d\boldsymbol{\beta}\ (Jensen's\ inequality) \\
&= E_q[log\ p(\vec{w_d}, \vec{t_d}, \vec{z_d}, \theta_d, \boldsymbol{\beta}, \boldsymbol{\tau}|\alpha, \eta)] - E_q[q(\theta_d, \vec{z_d}, \boldsymbol{\beta})]) \\
&= L(\vec{w_d}, \vec{t_d}, \boldsymbol{\gamma}, \boldsymbol{\lambda})
\end{aligned} \tag{5.2}$$

where the words of the document ($\vec{w_d}$) and their ($\vec{t_d}$) are observed variables. The log-likelihood of a document indicates the probability of observing this document given $\boldsymbol{\theta}$, $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$ (drawn from $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$ and $\boldsymbol{\rho}$). When Jensen's inequality (see Kuczma, 2009) is applied on the log-likelihood of the document, we can obtain a lower bound L (i.e. $L(\vec{w_d}, \vec{t_d}, \boldsymbol{\gamma}, \boldsymbol{\lambda})$) shown in the following equation:

$$\begin{aligned}
L(\vec{w_d}, \vec{t_d}, \boldsymbol{\gamma}, \boldsymbol{\lambda}) &= E_q[log\ p(\vec{w_d}, \vec{t_d}, \vec{z_d}, \theta_d, \boldsymbol{\beta}, \boldsymbol{\tau}|\alpha, \eta)] - E_q[q(\theta_d, \vec{z_d}, \boldsymbol{\beta})]) \\
&= E_q[log\ p(\vec{w_d}|\vec{z_d}, \boldsymbol{\beta})] + E_q[log\ p(\vec{z_d}|\theta_d)] \\
&\quad + E_q[log\ p(\theta_d|\alpha)] + E_q[log\ p(\boldsymbol{\beta}|\eta)] \\
&\quad + E_q[log\ p(\vec{t_d}|\vec{z_d}, \boldsymbol{\tau})] - E_q[q(\theta_d, \vec{z_d}, \boldsymbol{\beta})]
\end{aligned} \tag{5.3}$$

where $E_q[]$ means the expectation of the variational probability. Therefore, we can transform the process of minimising distance between the variational distribution and the true poste-

[7]The used variables are introduced in Section 5.2.2. They are all listed and described in Table 5.1

rior distribution into a process of maximising the lower bound of a document, L, which can be decomposed as shown in Equation (5.3). We apply expectation maximization (EM) for maximising L. Next, we first decompose the six summands in L and then introduce the EM approach for maximising L.

We decompose the six summands in L (see Equation (5.3)). The first two summands can be decomposed using the properties of the multinomial distributions:

$$\begin{aligned} E_q[log\ p(\vec{w_d}|\vec{z_d}, \boldsymbol{\beta})] &= \sum_{i,k} \phi_{d,i,k}\ E_q[log\ \beta_{k,i}] \\ E_q[log\ p(\vec{z_d}|\theta_d)] &= \sum_{i,k} \phi_{d,i,k} E_q[log\ \theta_{d,k}] \end{aligned} \tag{5.4}$$

where $\phi_{d,i,k}$ is the topic belief of a word $w_{d,i}$ (introduced in Section 2.2.3). We compute the expectation of the Dirichlet distribution and obtain the decomposed third and fourth summands as follows:

$$\begin{aligned} E_q[log\ p(\theta_d|\alpha)] &= log\Gamma(K\alpha) - Klog\Gamma(\alpha) + \sum_k (\alpha - 1) E_q[log\theta_{d,k}] \\ E_q[log\ p(\boldsymbol{\beta}|\eta)] &= log\Gamma(\sum_{i,k} \eta) - \sum_{i,k} log\Gamma(\eta) + \sum_{i,k} (\eta - 1) E_q[log\ \beta_{k,i}] \end{aligned} \tag{5.5}$$

where Γ is the Gamma function. To decompose the fifth summand, we apply the exception of the Beta distribution shown in Equation (5.6):

$$E_q[log\ p(\vec{t_d}|\vec{z_d}, \boldsymbol{\tau})] = \sum_{i,k} \phi_{d,i,k}((\rho_k^1 - 1) log\ t_{d,i} + (\rho_k^2 - 1) log\ (1 - t_{d,i})) \tag{5.6}$$

The last summand of the lower bound L, i.e. the log-expectation of the joint variational probability, is decomposed as shown in Equation (5.7):

$$\begin{aligned} E_q[q(\theta_d, z_d, \boldsymbol{\beta})] = &\sum_k E_q[log\ q(\theta_{d,k}|\gamma_{d,k})] \\ &+ \sum_i E_q[log\ q(z_{d,i}|\phi_{i,k})] \\ &+ \sum_{i,k} E_q[log\ q(\beta_{k,i}|\lambda_{k,i})] \end{aligned} \tag{5.7}$$

Finally, we have the expanded L shown in Equation (5.8):

$$\begin{aligned} L(\vec{w_d}, \vec{t_d}, \boldsymbol{\gamma}, \boldsymbol{\lambda}) = & \sum_{i,k} \phi_{d,i,k}\ E_q[log\ \beta_{k,i}] + \sum_{i,k} \phi_{d,i,k} E_q[log\ \theta_{d,k}] \\ & + log\Gamma(K\alpha) - K log\Gamma(\alpha) + \sum_{k} (\alpha - 1) E_q[log \theta_{d,k}] \\ & + log\Gamma(\sum_{i,k} \eta) - \sum_{i,k} log\Gamma(\eta) + \sum_{i,k} (\eta - 1) E_q[log\ \beta_{k,i}] \\ & + \sum_{i,k} \phi_{d,i,k}((\rho_k^1 - 1) log\ t_{d,i} + (\rho_k^2 - 1) log\ (1 - t_{d,i})) \\ & - \sum_{k} (\sum_{i} \phi_{d,i,k}\ log\ B(\rho_k^1, \rho_k^2)) \\ & - log\Gamma(\sum_{k} \gamma_k) + \sum_{k} log\Gamma(\gamma_k) - \sum_{k} (\gamma_k - 1) E_q[log \theta_{d,k}] \\ & - \sum_{i,k} \phi_{d,i,k} log \phi_{d,i,k} - log\Gamma(\sum_{i,k} \lambda_{k,i}) + \sum_{i,k} log\Gamma(\lambda_{k,i}) \\ & - \sum_{i,k} (\lambda_{k,i} - 1) E_q[log\ \beta_{k,i}] \end{aligned} \tag{5.8}$$

In EM, to maximise L, we first optimise $\phi_{d,i,k}$ by setting $\frac{\partial L_{\phi_{d,i,k}}}{\partial \phi_{d,i,k}} = 0$ (L is the lower bound, represented by Equation (5.8)) and then obtain the $\phi_{d,i,k}$ optimisation formula shown in Equation (5.9):

$$\begin{aligned} \phi_{d,i,k} \propto exp(& E_q[log \beta_{k,i}] + E_q[log \theta_{d,k}] \\ & + \delta((\rho_k^1 - 1) log\ t_{d,i} + (\rho_k^2 - 1) log\ (1 - t_{d,i}) \\ & - log\ B(\rho_k^1, \rho_k^2))) \end{aligned} \tag{5.9}$$

The classical VB approach (see Equation (2.8)) only has the *word statistics* (i.e. the first two summands $E_q[log \beta_{k,i}] + E_q[log \theta_{d,k}]$ in Equation (5.9)). Compared to the classical VB approach, the third summand in Equation (5.9) (i.e. $(\rho_k^1 - 1) log\ t_{d,i} + (\rho_k^2 - 1) log\ (1 - t_{d,i}) - log\ \ B(\rho_k^1, \rho_k^2)$, called the *time statistics*), is the additional feature we add to incorporate timestamps in our proposed TVB approach. Intuitively, the *time statistics* can have a direct impact on the term topic belief $\phi_{d,i,k}$. If a word $w_{d,i}$ is highly used in topic k at a time point t, $\phi_{d,i,k}$ is likely to be promoted if a tweet has the word $w_{d,i}$ with a timestamp t. However, the estimated shape parameters of the time distribution (i.e. $\boldsymbol{\rho}$) may not always fit a topic's trend well. An incorrectly estimated time distribution for a topic could give a negative bias on $\phi_{d,i,k}$. To solve this problem, we introduce a *balance* parameter δ, to control the impact of the *time statistics* on $\phi_{d,i,k}$ and alleviate such bias. Note that the influence of time in the ToT approach cannot be adjusted, e.g. through the δ parameter. Similar to $\phi_{d,i,k}$, we next

obtain the optimisation formula of γ and λ by setting their derivative of L to zero, shown in Equations (5.10) and (5.11) as follows:

$$\gamma_{d,i} = \alpha + \sum_{i,k} \phi_{d,i,k} \tag{5.10}$$

$$\lambda_{k,i} = \eta + \sum_{d,i,k} \phi_{d,i,k} \tag{5.11}$$

Meanwhile, to maximise L, we can also take the partial derivative with respect to the parameters of the Beta distribution, ρ_k^1/ρ_k^2. Actually, this step is equivalent to maximising the likelihood of the timestamps in topics. By optimising ρ_k^1/ρ_k^2, we also obtain the estimated topical trends. Taking the derivative to zero, we obtain the optimisation formula of ρ_k^1/ρ_k^2 shown in Equations (5.12) and (5.13):

$$\psi(\rho_k^1) - \psi(\rho_k^1 - \rho_k^2) = \frac{\sum_{d,i,k} \phi_{d,i,k} log\ t_{d,i}}{\sum_{d,i,k} \phi_{d,i,k}} \tag{5.12}$$

$$\psi(\rho_k^2) - \psi(\rho_k^1 - \rho_k^2) = \frac{\sum_{d,i,k} \phi_{d,i,k} log\ (1 - t_{d,i})}{\sum_{d,i,k} \phi_{d,i,k}} \tag{5.13}$$

where ψ is the Digamma function (log-derivative of Γ). Since ψ is involved in the optimisation equation, it is difficult to calculate ρ_k^1/ρ_k^2 directly. In our TVB approach, we estimate ρ_k^1/ρ_k^2 using a parameter optimisation algorithm[8].

5.2.3.2 Expectation Maximization Algorithm

Algorithm 1 shows the EM algorithm used by our TVB approach. In the iterative EM algorithm, we update ϕ and γ for each document (a tweet) in the E step. In the M step, λ and ρ_k^1/ρ_k^2 are updated using the statistics information (ϕ) from all posts. At the same time, all the timestamps are taken into account to estimate ρ_k^1/ρ_k^2. Figure 5.5 describes the update directions in the EM process between a classical VB approach and our TVB approach. In a classical VB approach (see Figure 5.5 (a)), the variational hyperparameters λ & γ (of the topic term distributions and the document topic distributions, respectively) are used to update ϕ for each document. In turn, λ & γ are then updated by ϕ. On the other hand, in TVB (see Figure 5.5 (b)), λ, γ and ρ together update ϕ in the E step. In the M step, ϕ updates λ, γ and ρ. The update direction from ρ is one of the main differences between the classical VB and our TVB approach.

[8]We simply apply a root finding algorithm (see Madsen, 1973) to estimate ρ_k^1/ρ_k^2.

Algorithm 1: Our TVB approach implemented by Expectation Maximization.

Initialize $\lambda_{N\times K},\ \gamma_{D\times K}$

while *L not converges* **do**

E step:

for $d\ <\ D$ **do**

repeat

for $i < N^d$ *&* $k\ < K$ **do**

$\phi_{d,i,k} \propto exp(E_q[log\beta_{k,i}] + E_q[log\theta_{d,k}]$
$+\delta((\rho_k^1 - 1)log\ t_{d,i} + (\rho_k^2 - 1)log\ (1 - t_{d,i})$
$-log\ B(\rho_k^1, \rho_k^2)))$ (i.e. Equation (5.9))

$\gamma_{d,k} = \alpha + \sum_{i,k} \phi_{d,i,k}$ (i.e. Equation (5.10))

until γ_d *converges*;

M step:

$$\psi(\rho_k^1) - \psi(\rho_k^1 - \rho_k^2) = \frac{\sum_{d,i,k} \phi_{d,i,k} log\ t_{d,i}}{\sum_{d,i,k} \phi_{d,i,k}}$$ (i.e. Equation (5.12))

$$\psi(\rho_k^2) - \psi(\rho_k^1 - \rho_k^2) = \frac{\sum_{d,i,k} \phi_{d,i,k} log\ (1 - t_{d,i})}{\sum_{d,i,k} \phi_{d,i,k}}$$ (i.e. Equation (5.13))

$\lambda_{k,i} = \eta + \sum_{d,i,k} \phi_{d,i,k},\ \forall i \in N$ (i.e. Equation (5.11))

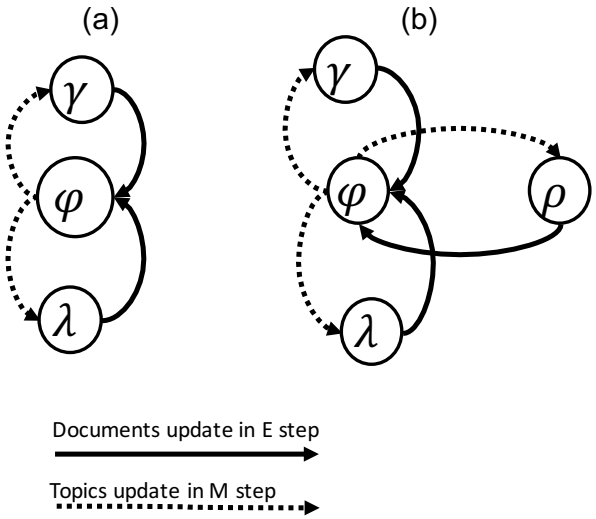

Figure 5.5: The update directions in the EM implementation in (a) the classical VB approach and (b) our TVB approach.

In this section, we have introduced how we implement our TVB approach. In the next section, we explain the main differences between our TVB approach and the related two topic modelling approaches in order to position our work.

5.3 Comparison to Baselines

Our TVB approach is not the first work to use the notion of time and to deal with tweets. In this section, we compare our TVB approach to the two most related topic modelling approaches: Topic Over Time (ToT) (c.f. Section 3.2.2.3) and Twitter LDA (TLDA) (c.f. Section 3.2.2.1). We discuss their main differences in this section.

5.3.1 Comparison to Topic Over Time (ToT)

Wang and McCallum (2006) first proposed the ToT topic modelling approach, which integrates the time dimension of news articles (introduced in Section 3.2.2.3). Although both ToT and TVB leverage the Beta distribution to integrate the time dimension, our proposed TVB approach builds on the variational inference approach while ToT is implemented using the sampling approach. There are three main differences between TVB and ToT. The first difference derives from the basic differences between the sampling and the VB approaches. Our approach is more efficient since the VB approach can be parallelised and can converge quicker than the sampling approach. Second, in ToT, the parameters ($\boldsymbol{\rho}$) of the topic time distributions are estimated using the method of moment (Bowman and Shenton, 2004), while these parameters are more intuitively estimated during the M step in our TVB approach. Indeed, in TVB, the $\boldsymbol{\rho}$ parameters are estimated during the EM process (see Equations (5.12) and (5.13)), where the estimation process at the same time also maximises the expectation in the E step (see Algorithm 1). Third, in TVB, we control both the *word statistics* and the *time statistics* (see Section 5.2.3) using the introduced *balance* parameter δ. However, in the ToT approach, the trade-off between the words and the time importance is not controlled. We argue that this might result in ToT generating mixed topics (introduced in Section 5.2.1). The reason is that a topic modelling approach is likely to group topics discussed in the same time period into the same generated topics, i.e. leading to mixed topics.

In our experiments, we use ToT as a baseline for evaluating our proposed TVB approach both in terms of the coherence of the generated topics and the extent to which they are mixed. Our hypotheses are that 1) TVB can generate topics with a higher coherence than ToT and 2) TVB can generate less mixed topics than ToT since the *time statistics* are controlled in TVB. We validate these two hypotheses in Section 5.5. In the next section, we compare TVB to another baseline, namely the Twitter LDA approach.

5.3.2 Comparison to Twitter LDA

Zhao et al. (2011b) deployed an additional Bernoulli distribution in Twitter LDA, which is used to control the background words (i.e. words shared by most of the topics) appearing in topics (discussed in Section 3.2.2.1). Intuitively, the background words have negative effects on generating coherent topics. By removing these background words, Zhao et al. (2011b) reported that the coherence of the generated topics can be improved. On the other hand, in TVB, we deploy a time distribution to model the timestamps of tweets. The time information integrated in TVB, naturally leads to further features being used in the modelling process, in addition to words. Hence, since our TVB approach leverages two types of features (words and tweet timestamps), our first hypothesis is that our TVB approach can generate topics with a higher coherence than Twitter LDA. Moreover, as discussed in Section 3.2.2.1, the single topic assignment strategy (which is applied in Twitter LDA) might lead to mixed topics. This is because the words used in tweets can belong to multiple themes. Indeed, aggressively assigning all words in a tweet into a single topic might cause the generation of mixed topics. Hence, our second hypothesis is that TVB is more effective than Twitter LDA in reducing the number of mixed topics.

Note that, in Twitter LDA, a user with their posted tweets are treated as one document. This strategy can work effectively if the Twitter users (e.g. writers, journalists, reporters.) have multiple tweets (see in Zhao et al., 2011b). However, we aim to apply Twitter LDA to extract topics during a political event on Twitter data, e.g. datasets about elections. In such datasets, the average number of tweets per user is rather low (this can be seen in Table 5.2). Hence, we postulate that Twitter LDA might not be as effective as TVB in extracting topics during a political event.

In the next sections, we evaluate the performance of Twitter LDA on two datasets, in comparison to TVB, ToT, as well as the classical LDA and the VB topic modelling approaches.

5.4 Experimental Setup

In this section, we evaluate our proposed TVB approach compared to 4 baselines from the literature, namely ToT, Twitter LDA (TLDA), and the classical sampling LDA (Gibbs) and VB approaches. Recall that our aim is to provide social scientists with a tailored topic modelling approach that generates topics from Twitter data during a political event. Such generated topics should be both easy for end-users to interpret and should not be mixed in order to further ease their interpretation. Hence, we evaluate the aforementioned 5 topic

modelling approaches in terms of their topic coherence as well as the extent to which they alleviate the generation of mixed topics from Twitter. Moreover, since the time dimension is used both in ToT and TVB, we evaluate these two approaches in terms of whether they can accurately estimate the real trends of topics. We argue that a good topic modelling approach for our end-users should (i) generate topics with a high coherence, (ii) generate topics that are less mixed, and (iii) accurately estimate the trends of the generated topics. Next, we describe our experimental setup for evaluating the 5 topic modelling approaches in terms of the above three aspects.

We first describe the used two real-world Twitter datasets in Section 5.4.1. Section 5.4.2 shows how we generate topics from the two used Twitter datasets using the 5 topic modelling approaches. We describe our used metrics in Section 5.4.3. Section 5.4.4 lists our four research questions.

5.4.1 Datasets

We collect two Twitter datasets for our experiments: 1) a ground-truth (GT) Twitter dataset and 2) a US Election 2016 (USE) Twitter dataset. The GT dataset is smaller and contains 8 known topics, which allows to evaluate the 5 topic modelling approaches using known ground-truth data (see Section 5.4.1.1). On the other hand, the USE dataset contains tweets posted during a major political event, i.e. the US Election 2016. Since we aim to apply topic modelling for a political event, this dataset allows us to specifically evaluate the effectiveness of the 5 compared approaches on a major political event. By using two datasets, we can examine the generalisation of the obtained results across both the GT and USE datasets. Next, we describe the details of the two used datasets in Sections 5.4.1.1 and 5.4.1.2, respectively.

Table 5.2: Two used Twitter datasets for evaluating the topic modelling approaches.

Dataset	Number of users	Number of tweets	Vocabulary size	Average tweets per user	Time period
GT	14,570	16,000	21,433	1.1	01/07/2016 - 31/08/2016
USE	40,296	79,167	26,646	2.0	01/08/2016 - 31/09/2016

5.4.1.1 Ground-Truth Dataset

The GT Twitter dataset contains 8 selected popular hashtag-events that occurred in July and August 2016. This dataset was collected using the Twitter API by searching for 8 hashtags: `#gopconvention`, `#teamgb`, `#badminton`, `#gameofthrone`, `#juno`, `#nba`,

#pokemongo and #theresamay. For each hashtag-event, we randomly sample 2,000 tweets, hence we obtain a Twitter dataset containing 16,000 tweets. As shown in Table 5.2, the GT dataset has $14.5k$ Twitter users posting 1.1 tweets in average. Such a ground-truth dataset has several advantages:

- The reasonable size (16K) of the Twitter corpus allows to efficiently conduct our experiments, i.e. all approaches can quickly converge.
- We avoid generating dominant and duplicated topics, thereby focusing the evaluation on the coherence quality of the topics.
- These predefined hashtags provide readily usable ground-truth labels, i.e. each hashtag-event is associated with the top 10 used words (labels of a topic) in its corresponding tweets. These labels of the 8 hashtag-events are used to match a generated topic with a hashtag-event. This enables us to evaluate how close the estimated topical trend is to its real trend (further details are given in Section 5.4.3.3).
- This ground-truth dataset allows humans assessors to effectively examine the generated topics and to conduct a user study described in Section 5.5. Indeed, since this dataset contains a limited number of topics, it is more feasible for human interpreters to evaluate all the generated topics of a given topic model in the conducted user study.

5.4.1.2 US Election 2016 Twitter Dataset

The USE Twitter dataset is related to major event in the US, namely the last US Election 2016 event. It contains tweets posted from 01/08/2016 to 31/09/2016. This dataset has about $40k$ users with $79.1k$ tweets (see Table 5.2) obtained by searching a list of US Election 2016-related keywords (e.g. "Trump", "Hillary", "Clinton" "debate", "vote", "election", etc) using the Twitter Streaming API. This method of creating a dataset is commonly used to collect event-related data from Twitter (e.g. in Vaccari et al., 2013; Sokolova et al., 2016). The collected USE dataset is different from the GT dataset in three main aspects. First, the USE dataset contains many more topics than the GT dataset, adding further complexities. Second, the topics in the GT dataset are balanced ($2k$ tweets per topic) while the topics in the USE dataset are not (e.g. some topics are discussed all thorough the event, while others are only discussed at very specific periods of time). Third, the USE dataset contains topics pertaining to a political event, while the GT dataset contains 8 diverse topics. As discussed earlier, the USE dataset permits to examine the performance of the 5 topic modelling approaches on a major political event, while the GT dataset allows us to validate their performance on known ground-truth labels.

In the next section, we explain how we apply the 5 topic modelling approaches on the two used Twitter datasets and the used metrics.

5.4.2 Generating Topics

For all approaches (Gibbs, TLDA, ToT, VB and TVB), η is set to 0.01 according to Blei et al. (2003); Griffiths and Steyvers (2004). We do not follow the traditional setting for α (α = 50/K), and set it instead to 0.4 for all approaches in our experiments, since in other separate preliminary experiments we noticed that a smaller α helps to generate topics with a higher coherence for short texts. The number of topics is set to 10 for the GT dataset, which is slightly higher than the real number of topics (8 in our dataset corresponding to 8 hashtags) because a slightly higher number of topics ensure that all hashtag-events can be extracted. For the USE dataset, given that it covers two months of tweets, we use K values that are likely to cover the many topics that could have been discussed by the Twitter users during that time period. In particular, we set the number of topics K to a lower-bound value of 50 and a upper-bound value of 100[9]. For all the sampling approaches (Gibbs, TLDA and ToT), we set the maximum number of iterations to 50. For the classical VB and our proposed TVB approaches, we set the number of iterations to 10 as the VB approaches converge more quickly. This setting allows us to verify whether the VB approach can generate high-quality topics with fewer iterations. Each experiment for each approach is repeated 10 times in order to conduct statistical significance using t-test, e.g. t-test is applied to identify whether 10 samples of topical coherence scores of topics from a topic model are significantly different from those of another topic model. In TLDA, recall that a document contains several tweets posted by a single Twitter user. However, most of the users in our Twitter dataset can have only one tweet, i.e. the average number of tweets per user is between 1 and 2 (see Table 5.2). Hence, to apply TLDA on our Twitter dataset, we create a virtual Twitter user by assigning 5[10] random tweets to this user. As mentioned in Section 2.2.1, in topic modelling, a document is seen as a mixture of topics. Therefore, randomly grouping tweets (having different topics) into a virtual user does not necessarily harm TLDA (as will shown in the results). For all the other approaches (TVB, ToT, Gibbs and VB), a document represents a single tweet. For our TVB approach, we vary the *balance* parameter (discussed in Section 5.2.3) δ=$\{$0.4, 0.6, 0.8, 1.0$\}$ to evaluate how it impacts performance when generating coherent topics.

[9]We do not set many different values of K, since we aim to evaluate the 5 topic modelling approaches in this chapter.

[10]Naturally, we can use any other values. However, we found that 5 tweets are sufficient to create a virtual document covering more than 1 topic.

5.4.3 Evaluation Metrics

We apply the topic coherence metrics (introduced in Chapter 4) to automatically evaluate the coherence level of the generated topics (see Section 5.4.3.1). We introduce a topic mixing degree metric (see Section 5.4.3.2), which indicates the extent to which the generated topics are mixed together. Since both the ToT and TVB approaches estimate the topical trends, we also use the trend estimation error (see Section 5.4.3.3) to compute the distance between a real topical trend and its estimated topical trend (introduced in Section 5.2.1).

5.4.3.1 Metric 1: Coherence Metrics

Following the conclusion provided in Chapter 4, we use the T-WE coherence metric (i.e. Metric (18) in Table 4.6) to evaluate the coherence of the generated topics. In order to capture the semantic similarity of the latest hashtags and Twitter handle names, we crawl 200 million English tweets posted from 01/08/2015 to 30/08/2016 using the Twitter Streaming API. This Twitter dataset is crawled in a different time period compared to the Twitter background dataset used in Chapter 4 (see Section 4.5.1). Indeed, the time period of this newly crawled Twitter background dataset covers the time period of the GT dataset as well as a 13-month time period before the US Election 2016 date (i.e. 08/11/2016). Therefore, using the new background dataset, T-WE can effectively assess the coherence of the topics generated from both the GT and USE datasets. We train a WE model using this Twitter background dataset and obtain word embedding vectors of 5 million tokens[11]. The trained WE model is used in our WE-based coherence metric. To evaluate the coherence of topic models, we apply our proposed coherence@n metric (denoted as `c@n`, see Section 4.6). Note that the `c@n` metric calculates the average coherence scores of the top n ranked topics in a topic model (introduced in Section 4.6), where the coherence of topics are computed using the WE-based coherence metric. For the GT dataset, we examine the top 2 and 7 most coherent topics from a generated topic model, i.e. `c@2` & `c@7` metrics. Considering that the number of topics is 10, we argue that the top 2 and 7 most coherent topics are reasonable choices to evaluate the coherence of the generated topic models. For the USE dataset, we use `c@10` & `c@20` and `c@30` metrics as the number of topics is relatively bigger. We also apply the average (`Aver`) coherence to evaluate all topics for both Twitter datasets, i.e. the average coherence score of all topics in a topic model (recall that `Aver` is used as a baseline for `c@n` in Section 4.6)

[11]The method used to train the WE model is the same as the method described in Section 4.5.2.3.

5.4.3.2 Metric 2: Mixing Degree Metric

As discussed in Section 5.2.1, Topic (a) ("*currency, GBP, weaker*") can be mixed with Topic (b) ("*currency, Scotland, euro*") because they have a similar usage of words, such as the use of "*currency*" in Figure 5.3. Let's assume that Topic (c) is represented by "*scotland, economy, finance* ". Topic (b) can also be mixed with Topic (c) since Topic (c) has the words "*scotland, economy*", which are semantically related to "*scotland, currency*". Topics (a), (b) and (c) are mixed, in the sense that they have overlapping/related topics. We introduce a new metric to capture the similarities of all pairs of topics generated by a given topic modelling approach. The higher the overall similarity, the more mixed are the generated topics. More formally, we use cosine similarity to compute the average similarities among all the generated topics, which we call the *topic mixing degree* (denoted as `MD`). We use Equation (5.14) to calculate the `MD` score of a topic model:

$$MD(\boldsymbol{\beta}) = \sum_{k} \sum_{k'} cosine(\beta_k, \beta_{k'})/|K|^2 \tag{5.14}$$

where β_k is a topic term distribution and K is the total number of topics (see Table 5.1). The higher ***MD*** is, the more the topic model is mixed, i.e. the topic modelling approach generated more mixed topics. A similar methodology is used in AlSumait et al. (2009) to identify the background topics.

5.4.3.3 Metric 3: Trend Estimation Error

Both the ToT and our TVB approaches estimate the topical trends. To evaluate the topical trends over time, we calculate the distance/error between the real topic trends and the estimated topical trends (using the Beta distribution in ToT and TVB). The error is calculated using the method shown in Equation (5.15):

$$ERR(\tau) = \frac{\sum_k \int_0^1 |\tau_k(t) - PDF_k(t)|dt}{K} \tag{5.15}$$

where $PDF_k(t)$ is the probability density function of the real timestamps of topics, which is obtained through the GT dataset. The `ERR` score ranges from 0 to 2. The generated topics are matched to the ground-truth topics if the top 10 words of a generated topic have at least 3[12] same words to the top 10 words of a hashtag event.

For the GT dataset, we apply the three mentioned metrics: topic coherence metrics (`c@n` and `Aver`), topic mixing degree metric (`MD`) and trend estimation error metric (`ERR`).

[12] 3 mutual words in the top 10 words is a reasonable minimum number to indicate a similar topic.

However, there are no ground-truth labels in the USE dataset. Hence, only c@n, Aver and MD metrics are used for the USE dataset.

5.4.4 Research Questions

We aim to answer four research questions in this chapter:

- **RQ1.** Does our TVB approach outperform ToT and TLDA in terms of topic coherence and topic mixing degree?
- **RQ2.** Does the time dimension help to improve the coherence of topics in our TVB approach?
- **RQ3.** What is the impact of the *balance* parameter on both the coherence and the mixing degree of the generated topics?
- **RQ4.** Does our TVB approach more accurately estimate the trends of the generated topics compared to ToT?

5.5 Evaluation of TVB

In this section, we first analyse the performance of the 5 topic modelling approaches (Gibbs, TLDA, ToT, VB and TVB) using the GT Twitter dataset in terms of the c@n, Aver, MD and ERR metric (Section 5.5.1). Then we report the performance of the 5 topic modelling approaches on the USE Twitter dataset in terms of c@n, Aver and MD metrics in Section 5.5.2. We summaries the results from the two datasets and answer the four research questions in Section 5.5.3. Finally, we discuss the efficiency of the 5 topic modelling approaches in Section 5.5.4.

5.5.1 Results and Analysis on the GT Dataset

Table 5.3 shows the obtained results for the GT dataset. The listed scores are the average scores of 10 models (each approach is repeated 10 times, see Section 5.4.2) generated by each approach with respect to the 3 types of metrics (described in Section 5.4.3). For the coherence metrics, Aver, c@2 and c@7, a higher score corresponds to more coherent topics, whereas lower scores for the MD and ERR metrics indicate higher quality models. The subscripts indicate whether a given approach is significantly ($p<0.05$, t-test, see Section 5.4.2) better than the other ones. For example, the Aver score of TVB (T') with $\delta = 0.8$, 0.158_v, is significantly better than that of the VB approach, indicating that TVB generates topics with

a higher coherence than VB. To help understand the topical trends, we randomly choose one model from each of the ToT and TVB models and list their estimated topical trends together with the real trends in Figure 5.6. Next, we first analyse the results in terms of the topical coherence and topical mixing degree metric. We then report the results of our conducted user study to verify our mixing degree metric. Then, we discuss the performance of ToT and TVB in estimating topical trends in terms of the estimation error (ERR).

Table 5.3: The topic coherence, mixing degree and topic trends estimation error of the topic modelling approaches on the **GT** dataset. The subscripts indicate whether a given approach is significantly ($p<0.05$, using t-test) better than the other ones. The bold font indicates the highest value for each column.

Models	Coherence			MD	ERR
	Aver	c@2	c@7		
Gibbs (G)	0.154	0.204	0.168	$0.051_{W,T}$	$\times$
TLDA (W)	$\mathbf{0.177}_{G,V,T,T'}$	$\mathbf{0.248}_{G,V,T,T'}$	$\mathbf{0.198}_{G,V,T,T'}$	0.102_{T}	$\times$
VB (V)	0.151	0.201	0.165	$0.049_{W,T}$	$\times$
ToT (T)	$0.160_{G,V}$	0.205	0.175_{V}	0.149	1.358
TVB(T'), $\delta = 0.4$	0.152	0.202	0.165	$0.043_{W,T}$	1.211_{T}
TVB(T'), $\delta = 0.6$	0.153	0.204	0.166	$\mathbf{0.042}_{W,T}$	1.256_{T}
TVB(T'), $\delta = 0.8$	0.158_{V}	$0.221_{G,V,T}$	0.174_{V}	$0.047_{W,T}$	1.206_{T}
TVB(T'), $\delta = 1.0$	0.156_{V}	0.209	0.170	$0.055_{W,T}$	$\mathbf{1.168}_{T}$

5.5.1.1 Topical Coherence and Topical Mixing Degree on the GT Dataset

The 5 topic modelling approaches, Gibbs, TLDA, ToT, VB and TVB, are denoted as G, W, V, T and T' in Table 5.3. First, for the topical coherence, it is clear that TLDA (W) performs best and significantly outperforms all of the other approaches on the GT dataset. Our TVB approach (TVB with δ=0.8) performs second best since the coherence of the generated topics by TVB is higher than those of Gibbs, TLDA and VB, indicated by the Aver, c@2 and c@7. On one hand, these results suggest that the hypothesis (see Section 5.3.1) that our TVB approach generates topics with a higher coherence than TLDA does not hold on the GT dataset. The first reason could be that the single assignment method is more effective than the use of time in TVB for generating coherent topics. Another reason might be that TLDA employs a distribution to control the use of background words in topics (see Section 5.3.2), which could help to improve the coherence of topics from tweets. Since we observe that TVB is statistically significantly better than VB in terms of topical coherence, we can conclude that integrating time into VB is effective. Therefore, it is more likely that TLDA benefited from the additional control of the distribution of background words to outperform TVB. On the other hand, our hypothesis (see Section 5.3.2) that our TVB approach generates topics

with a higher coherence than ToT appears to hold on the GT dataset. There are no significant differences between the ToT and TVB models in terms of c@7 and Aver. However, we observe that TVB performs significantly better than ToT, Gibbs and VB on the top 2 most coherent topics, as suggested by c@2. Indeed, we can see the positive impact of the time dimension in improving the coherence of models in both TVB and ToT. For example, the Aver scores of ToT models are significantly better than those of both Gibbs and VB models while the c@2 scores of TVB (with $\delta = 0.8$) models are significantly better than those of the Gibbs and VB models. We also observe that our TVB models with $\delta = 0.8$ perform better than the TVB models with a lower/higher δ (TVB with $\delta = 0.4, 0.6/1.0$). This indicates that alleviating the bias of the *time statistics* (described in Section 5.2.3) helps to generate topics with a higher coherence, which suggests that the *balance* parameter δ has a positive impact in our TVB approach. In summary, on the GT dataset, while TLDA performs best, TVB performs second-best and ToT performs comparably to TVB in terms of topical coherence.

In terms of the MD metric, our TVB models (with $\delta = 0.6$) have the lowest MD scores. In particular, all TVB models have significantly lower MD scores than the TLDA and ToT models. This suggests that our hypotheses that TVB can generate less mixed topics than ToT and TLDA seems correct. Among the 5 approaches, ToT models have the highest MD scores indicating ToT models have topics that are highly mixed. This could be that the *time statistics* in ToT are not controlled and ToT is likely to treat the topics happening in the same time period into a single topic (i.e. a mixed topic). In our TVB model, when we increase the value of δ (the *balance* parameter), the MD score of a TVB model increases, for example, the MD of TVB with ($\delta = 1$) is higher than that of TVB with ($\delta = 0.6$), which means that the topics are more mixed when the importance of *time statistics* increases. This supports the reason why ToT generates mixed topics, i.e. the importance of *time statistics* is not controlled in ToT. Moreover, it again shows the importance of the *balance* parameter δ in TVB. On the other hand, although TLDA models have the highest coherence scores, they have rather high MD scores just after ToT, which also suggests that TLDA is likely to generate mixed topics. There can be two reasons. First, as discussed in Section 5.3.2, TLDA applies the single topic assignment strategy, where all words in a tweet are assigned with the same topic. This single topic assignment strategy can introduce mixed topics in TLDA since the words in a tweet can be discussed in multiple topics. Second, we randomly assign 5 tweets into a virtual user for TLDA, which could cause mixed topics. The classical Gibbs and VB approaches do not have high MD scores on the GT dataset in our experiments.

Overall, TLDA has a rather high mixing degree (MD), indicating that its generated topics are mixed, although these topics appear to be coherent. On the other hand, our TVB approach generated less mixed topics but their coherence was lower than TLDA. To validate

our conclusion that TVB does indeed outperform TLDA in generating less mixed topics, we conduct a user study comparing TVB with TLDA[13] in terms of how likely the topics they generated are mixed. The user study also allows us to evaluate the extent to which our proposed MD metric is aligned with human judgements.

5.5.1.2 A User Study of Mixed Topics

We choose to compare the mixing degree between the TVB with $\delta = 0.8$ and TLDA models using human judgements. In our user study, we ask 8 expert end-users[14] whether a given topic contains multiple themes. Specifically, both the TVB and TLDA approaches generate 10 models. We pair these 20 models randomly and generate 10 pairs, where each pair has one model (containing 10 topics) from TVB and another one (i.e. 10 topics) from TLDA. For each pair, we present a human with all the generated topics of the 2 models. Before the task, we present the 8 users with the basic knowledge of the 8 topics (see Section 5.4.1.1). A human assessor is asked to identify all of the multi-theme topics from 2 given models (10 topics per model). A model in a pair is preferred (i.e. obtains a vote), if a human assessor finds less multi-theme topics in this model pair. Each pair gets 3 judgements from 3 different humans. An approach obtains a credit if its model in a pair obtains a majority from the 3 votes.

Table 5.4 lists the number of credits and votes obtained by TVB and TLDA, respectively. We observe that our TVB approach obtains 7 credits while the TLDA approach only obtains 2 credits. Out of the 10 model pairs, the human assessors did not agree on one of them. Among the 10 pairs, TVB obtains all assessors' votes in 5 pairs (i.e. 15 votes), 2 votes in 2 pairs (i.e. 4 votes) and 1 vote in the remaining 3 pairs (i.e. 3 votes). In total, TVB obtains 22 votes. On the other hand, TLDA obtains only 7 votes from human assessors. These results suggest that our TVB models have less mixed topics than the TLDA models.

Table 5.4: The results of our user study on mixed topics.

Dataset	TVB with $\delta = 0.8$	TLDA
Number of obtained credits	7	2
Number of obtained votes	22	7

To better understand the topics in our user study, we list two topic examples of our TLDA and TVB ($\delta = 0.8$) models in Tables 5.5 and 5.6. Both models generate human interpretable topics. However, we observe more multi-theme topics in the TLDA models, such as

[13]The MD scores of the ToT models are significantly higher than both the TVB and TLDA models. Hence, we do not include them in our user study. Indeed, if the human assessors find that TLDA generates more mixed topics than TVB, then it is reasonable to conclude that ToT also generates more mixed topics than TVB.

[14]Members of the Terrier (http://terrierteam.dcs.gla.ac.uk) research team.

Table 5.5: Topic samples from a TLDA model on the GT dataset, where the underlined words have a different topic theme from the rest of words in a topic. Note that we present a human assessor with the top 10 words of a topic in our user study. In this table, we only list the top 5 words for each topic.

Topic	TLDA
1	#rio #badminton #olympics #iamteamgb wei
2	#jupiter #juno @nasa orbit @nasajuno
3	#nbasummer nba #basketball @nba basketball
4	@gameofthrones #emmys season outstanding
5	#rncincle trump speech melania donald
6	#rio #badminton #iamteamgb team gold
7	#iamteamgb #theresamaypm thanks #jupiter
8	thrones game pokemon season like #pokemon

Table 5.6: Topic samples from a TVB ($\delta = 0.8$) model on the GT dataset, where the underlined words have a different topic theme from the rest of words in a topic. Note that we present a human assessor with the top 10 words of a topic in our user study. In this table, we only list the top 5 words for each topic.

Topic	$\text{TVB}_{\delta=0.8}$
1	#badminton #rio #mas #olympics wei chong
2	#juno burn engine complete unlock #jupiter
3	nba #basketball sign wire basketball
4	thanks @gameofthrones #iamteamgb #emmys
5	#rncincle trump @realdonaldtrump speech
6	#iamteamgb win medal #rio @teamgb
7	#theresamaypm watch #brexit minister prime
8	pokemon basketball team usa #pokemon news

"badminton"(topic 1), "teamgb" (topic 6), "theresamaypm" (topic 7) and "pokemon"(topic 8), while the TVB model has less multi-theme topics: "gameofthrone" (topic 4) and "pokemon"(topic 8). In fact, it is easy to mix the topics "theresamaypm" and "teamgb" since they are all popular topics in the UK, and it is possible that the word usage in these two topics is similar. However, the topical trends of these two topics are not similar: "theresamaypm" was popular around 11/07/2016 when Theresa May became the new UK Prime Minister, while "teamgb" was highly discussed during the Olympic Games (from 05/08/2016 to 21/08/2016) (see the topical trends in Figure 5.6).

Our user study shows that our TVB models do indeed have less mixed topics than the TLDA models, as judged by human assessors. Importantly, our user study does show that our MD metric is aligned with human judgements.

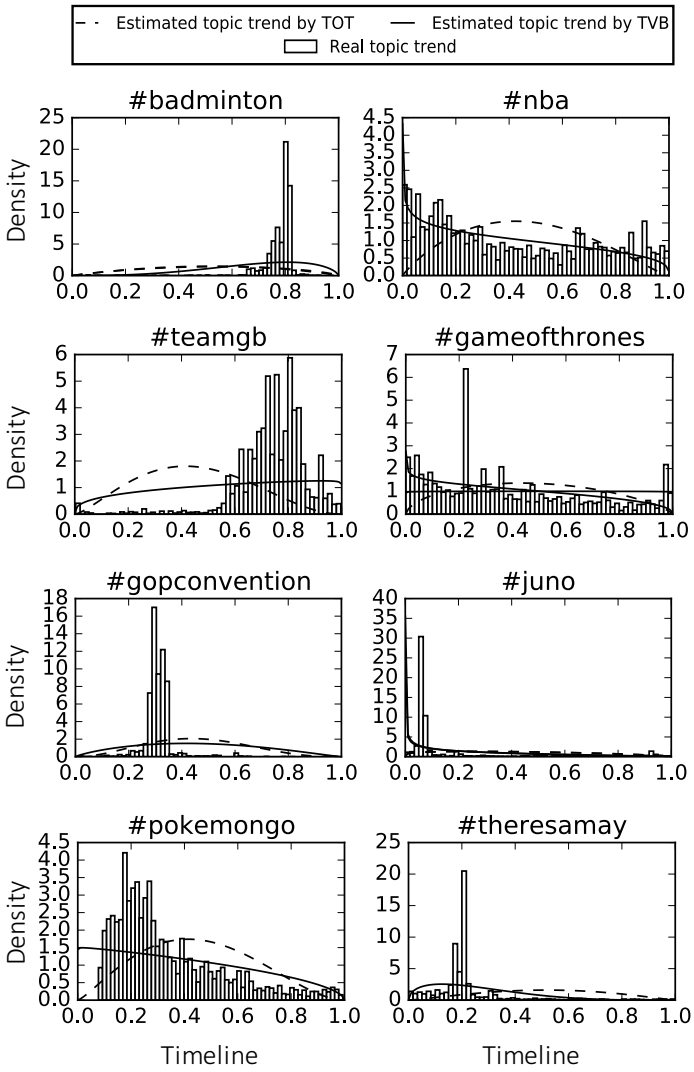

Figure 5.6: The real and estimated topical trends estimated by the two topic modelling approaches on the GT dataset, where the x-axis and the y-axis represent the timeline and the density probability, respectively.

5.5.1.3 Topical Trends Estimation Error on the GT Dataset

Both the ToT and TVB approaches estimate topical trends. The ERR metric indicates the distance between the real topical trends and the estimated ones. Smaller distances are better. The ERR scores in Table 5.3 suggest that our TVB approach generates significantly (p <0.05 using the t-test) more accurate topical trends than the ToT approach. The main reason can be that the ToT approach has a very high mixing degree (discussed in Section 5.5.1.1). The estimated trend of a mixed topic is different from the real trend of a topic and therefore the ToT approach has a significantly higher ERR score. Unlike the ToT and TLDA approaches, our TVB model has less multi-theme topics, which results in a more accurate estimation of the

topical trends. In Figure 5.6, we list the estimated trends of topics from a TVB ($\delta = 0.8$) and a ToT models, represented by solids line and dashed lines, respectively. They are compared to the real trends of the 8 topics (i.e. histograms in Figure 5.6). Both chosen models have duplicated topics, which are `#badmintion` & `#juno` and `#gameofthrone` & `#juno` in the ToT and TVB models, respectively. Since the ToT models have more multi-theme topics, it is difficult to match the generated topics with the real ones. For example, the topic theme `#nba` is mixed with `#pokemongo` in the ToT model. As a result, the estimated trend of ToT for `#nba` is not accurate. Although both the ToT and TVB models do not exactly fit the real topical trends using Beta distributions, it is still clear that the estimated trends from the TVB model are closer to the real trends than those of the ToT model, i.e. the solid lines (trends estimated by TVB) are closer to the histograms compared to the dashed lines (trends estimated by ToT) as illustrated in Figure 5.6.

In this section, we have analysed the results of the 5 topic modelling approaches on the GT dataset. In the next section, we compare the 5 topic modelling approaches on the USE dataset.

5.5.2 Results and Analysis on the USE Dataset

Unlike the GT dataset, the topics in the USE dataset are unknown. It is not possible to compare the real topical trends to the estimated trends by the ToT and TVB approaches. Hence, we only report results of the 5 topic modelling approaches in terms of the `c@n`, `Aver` and `MD` metrics for the USE dataset. Tables 5.7 and 5.8 show the `c@n`, `Aver` and `MD` scores of the 5 types of topic models with $K = 50$ and $K = 100$, respectively. These tables use the same notations as Table 5.3. In the previous section, we showed that our TVB approach obtains the best performance when the *balance* parameter δ is set to 0.8. Therefore, we use this setup for the USE dataset. We expect to see that δ = 0.8 will work effectively in the USE dataset.

First, while TLDA significantly outperforms the other 4 approaches in terms of topical coherence in the GT dataset, we do not observe that TLDA performs better than the other 4 approaches in the USE dataset when K is set 50 or 100. The reason can be that the USE dataset has more noise than the GT dataset. TLDA could not effectively deal with a much noisier dataset. On the other hand, we find that TVB generates topics with a higher coherence than Gibbs, VB and TLDA. In particular, the top 30 topics from our TVB models have higher coherence than those from TLDA (significantly), Gibbs(significantly), VB (significantly) and ToT (not significantly) models suggested by `c@30` when $K = 50$ and $K = 100$ (see Tables 5.7 and 5.8). Moreover, the coherence of the top 10 topics of TVB is also higher

than those of the rest of topic models shown in both Tables 5.7 and 5.8. Indeed, we observe that TVB performs best in the USE dataset considering that TVB has clear advantage in generating top 10 and top 30 topics with a higher coherence than the other approaches. This again indicates that the use of the time dimension and the use of the *balance* parameter are effective in TVB. In addition, ToT performs slightly better than TLDA since it has a better `Aver` scores. The `c@n` scores of both ToT and TVB models are close. Similar to the GT dataset, the Gibbs and VB models are not as good as the others since most of their `Aver` and `c@n` scores are low.

Table 5.7: The topical coherence and mixing degree of the 5 topic modelling approaches on the **USE** dataset with $\boldsymbol{K}$ **= 50**. The subscripts indicate whether a given approach is significantly (p<0.05, using t-test) better than the other ones. The highest score in each column is in bold.

Models	Coherence				MD
	Aver	c@10	c@20	c@30	
Gibbs (G)	0.212	0.298^{V}	$\mathbf{0.267}^{V}$	0.235^{V}	0.113^{T}
TLDA (W)	0.210	0.297^{V}	0.266^{V}	0.235^{V}	0.091^{T}
VB (V)	0.196	0.259	0.237	0.222	$\mathbf{0.07}^{G,W,T}$
ToT (T)	$\mathbf{0.225}^{V}$	0.294^{V}	0.266^{V}	0.239^{V}	0.210
TVB(T'), $\delta = 0.8$	0.214	$\mathbf{0.300}^{V}$	0.264^{V}	$\mathbf{0.243}^{G,W,V}$	$0.079^{G,W,T}$

Table 5.8: The topical coherence and mixing degree of the 5 topic modelling approaches on the **USE** dataset with $\boldsymbol{K}$ **= 100**. The subscripts indicate whether a given approach is significantly (p<0.05, using t-test) better than the other ones. The highest score in each column is in bold.

Models	Coherence				MD
	Aver	c@10	c@20	c@30	
Gibbs (G)	0.199	0.283_{V}	0.278_{V}	0.249_{V}	0.087^{T}
TLDA (W)	0.201_{V}	0.297_{V}	$\mathbf{0.286}_{V}$	0.259_{V}	0.079^{T}
VB (V)	0.185	0.250	0.249	0.236	$0.066^{G,W,T}$
ToT (T)	$\mathbf{0.203}_{V}$	0.297_{V}	0.281_{V}	0.258_{V}	0.144
TVB(T'), $\delta = 0.8$	0.191	$\mathbf{0.298}_{V}$	0.283_{V}	$\mathbf{0.277}_{G,W,V,T}$	$\mathbf{0.057}^{G,W,T}$

Second, we observe that both TVB and VB have clear advantage in generating less mixed topic on the USE dataset. Both TVB and VB models have significantly lower `MD` scores compare to the other 3 types of topic models, as can be seen in Tables 5.7 and 5.8. These results are similar to the results of the GT dataset, which suggests that our TVB approach generates less mixed topics than TLDA, ToT and Gibbs. On the other hand, ToT still performs worse in terms of `MD` on the USE dataset. Third, we observe that TVB with δ = 0.8 works effectively in our USE dataset since it helps TVB to generate topics with higher

coherence. Hence, we recommend to set $\delta = 0.8$ in TVB when extracting topics from Twitter data.

So far, we have reported and analysed the performance of the 5 topic modelling approaches on two Twitter datasets. Next, we summarise the results and answer the four research questions listed in Section 5.4.4.

5.5.3 Summary of Results

In the previous sections, we used two Twitter datasets to evaluate our proposed TVB approach compared to 4 baseline approaches: Gibbs, TLDA, ToT and VB. First, we address the first research question (**RQ1** in Section 5.4.4). In terms of topical coherence, on the GT dataset, TVB performs second-best overall and outperforms ToT (see the `c@2` score in Table 5.3). However, TVB does not outperform TLDA, the most effective topic modelling approach according to our reported results on the GT dataset. On the other hand, our TVB approach outperforms both TLDA (significantly) and ToT (not significantly) on the USE dataset. In addition, in terms of the `MD` metric, our TVB approach generated significantly less mixed topics than TLDA and ToT across both used Twitter datasets. Note that TVB performs particularly effectively on the USE dataset pertaining to a major political event, unlike GT where the topics are not necessarily related to political events.

The obtained results above suggest that there is no clear winner between TVB and TLDA in terms of topical coherence. However, it is clear that TVB generates less mixed topics than all other 4 baselines. Overall, in answer to **RQ1**, TVB does outperform ToT and TLDA on the USE political event dataset but not on the more diverse GT dataset. Nevertheless, overall, TVB does appear to be a promising and effective approach for generating coherent and interpretable topics for a political event on Twitter.

Next, we answer the second question (**RQ2** in Section 5.4.4). The obtained results demonstrate that TVB outperforms VB on both the GT and USE datasets. Hence, the time dimension appears to help (significantly) improve the coherence of the generated topics while alleviating their mixing degree (not significantly).

For the third research question (**RQ3** in Section 5.4.4), our results on the GT dataset indicate that a properly set *balance* parameter (δ) can help to enhance both the coherence and the mixing degree of the generated topics. Overall, TVB does generate topics that are coherent and less mixed (see the rows with different values of δ in Table 5.3). Finally, in answer to the fourth research question (**RQ4** in Section 5.4.4), our results on the GT dataset demonstrate that TVB can estimate the trends of the generated topics significantly more accurately than ToT (see the `ERR` column in Table 5.3). Thus far, we have investigated the

effectiveness of the 5 compared topic modelling approaches. In the next section, we report their efficiency performance.

5.5.4 Efficiency of the five Topic Modelling Approaches

We record the time consumption of the five used topic modelling approaches in order to examine their efficiency. All these topic modelling approaches are implemented in Python. Specifically, Gibbs and VB are implemented using an open source code[15]. We mapped the original TLDA Java code[16] and the original ToT Java code[17] into Python versions for a fair comparison. We use a machine with Intel Core i7 (3.59 GHz) and 16GB RAM to conduct our experiments. Although the VB and TVB approaches can be implemented in parallel, we do not apply parallel computation (i.e. using multiple threads) because this is not our focus in this thesis. We apply the 5 approaches on the USE dataset since this dataset has more tweets ($79.2k$) and this allows us to know the efficiency of these approaches on a dataset with a larger size. For ToT and TLDA, we set the number of iterations to 50 and we set to 10 for VB and TVB. We record the time points when the 5 approaches start and finish the process of generating topics using single thread processing and then compute the average consumed time per iteration.

Table 5.9: The time consumption of the 5 topic modelling approaches on the USE dataset.

	Gibbs	TLDA	VB	ToT	TVB
Average consumed time per iteration (second)	**15.56**	70.96	31.27	86.01	68.83

Table 5.9 shows the consumed time for the USE Twitter datasets in Table 5.9, where the consumed time is the average consumed time of 5 repeating experiments per iteration. It is obvious that the Gibbs (sampling approach) and the VB approach are the most efficient approaches. Other enhancements (TLDA, ToT and TVB) take a longer time for computing each iteration compared to Gibbs and VB. However, to generate a topic model with good quality, our TVB approach spends less time than TLDA and ToT. This suggests that our TVB approach is effective and also reasonably efficient.

[15] https://github.com/dongwookim-ml/python-topic-model
[16] https://github.com/minghui/Twitter-LDA
[17] https://github.com/ahmaurya/topics_over_time

5.6 Conclusions

In this chapter, we have proposed a time-sensitive topic modelling (TVB) approach, which can be used to address 'what' has been discussed on Twitter. Our TVB approach generated topics from tweets taking into account 'when' the topics were discussed and therefore can effectively identify and distinguish the discussed topics on Twitter. Our proposed TVB approach, which extends the classical Variational Bayesian approach (Section 2.2.3), employed the Beta distribution to integrate time, where the importance of *time statistics* were controlled by a *balance* parameter (c.f. Section 5.2.3). To evaluate our TVB approach together with the other four topic modelling approaches, i.e. the classical Gibbs sampling, the classical VB, the Twitter LDA (TLDA) and the Topics over Time LDA (ToT) approaches, we used a ground-truth (GT) Twitter dataset and a US Election (USE) Twitter dataset to conduct our experiments. We compared the five topic modelling approaches using metrics of the topic coherence, topic mixing degree and trend estimation error (see Section 5.4.3). We showed that the time dimension helped to generate more coherent topics in our TVB models across the two Twitter datasets, compared to the baseline approaches, such as the classical Gibbs sampling and VB approaches (see Tables 5.3, 5.7 and 5.8). We demonstrated that our TVB approach performed second-best in GT dataset and best in the USE dataset in terms of topical coherence, which indicated the high effectiveness of our TVB approach. Moreover, we showed that our TVB approach significantly generates less mixed topics compare to two main baselines, i.e. ToT and TLDA. In addition, our TVB approach is significantly better than ToT when estimating the trends of the generated topics. We concluded that our TVB approach was overall promising and effective when generating coherent and human interpretable topics for Twitter data.

So far, we have investigated an effective time-sensitive topic modelling approach to address the 'what' taking into account the importance of the time (i.e. 'when') for Twitter data. To identify 'who' participated in the topic discussions, we turn to investigate the Twitter user community classification in the next chapter.

Chapter 6

Twitter User Community Classification

6.1 Introduction

In the previous chapters, we have investigated various topic coherence metrics and a new tailored Twitter topic modelling approach, which addressed 'what' was said and 'when' during a political event through the generation of highly coherent topics from Twitter data. In this chapter, we aim to identify 'who' was involved in a political event, i.e. which community a Twitter user belongs to. To do so, we study the identification of communities on Twitter. As discussed in Section 3.4, prior work has investigated how to automatically categorise Twitter users. However, there are still limitations when classifying Twitter users into communities (highlighted in Section 3.4). To conduct the Twitter user community classification task, there are no suitable automatic ground-truth generation approaches. Moreover, it is not clear how the classifiers, trained using the automatically generated dataset, perform when classifying Twitter users into communities. In this chapter, we investigate two aspects of Twitter user community classification: ground-truth generation and classification approaches.

To obtain ground-truth data for learning effective user community classifiers, we propose two automatic ground-truth generation approaches: a hashtag labelling approach and a DBpedia[1] labelling approach. The hashtag labelling approach can be used to identify the communities with different political orientations while the DBpedia labelling approach can identify the communities in terms of the Twitter users' professions (e.g. business elites and academics). The hashtag labelling approach leverages how Twitter users use the hashtags that are indicative of their political orientations during an election or a referendum. To validate the ground-truth data generated using the hashtag labelling approach, we examine the followee network of Twitter users. On the other hand, the DBpedia labelling approach labels

[1] `https://wiki.dbpedia.org`

the Twitter users by leveraging how the community-related keywords (e.g. 'professor' for the academic community) are used in their profiles and tweets. To evaluate the DBpedia labelling approach, we compare the generated labels to the human ground-truth labels, which are obtained through a user study. In developing our Twitter user community classification task, we use the ground-truth datasets generated using the hashtag and the DBpepdia labelling approaches for training the Twitter users community classifiers. To effectively classify the community affiliations of Twitter users, we propose a Topic-based Naive Bayes (TBNB) approach tailored to Twitter, which identifies the community affiliations by considering the word usage in their tweets in both the discussed topics and the identified communities. We evaluate our TBNB approach by conducting experiments using ground-truth datasets generated using the two proposed ground-truth generation approaches. We show that by using the discussed topics, our proposed TBNB approach can better identify the community affiliations of Twitter users compared to the commonly used baseline classifiers, such as Naive Bayes. The remainder of this chapter is organised as follows:

- Section 6.2 introduces two ground-truth generation approaches. We also evaluate these ground-truth generation approaches and use them to generate the ground-truth datasets for our Twitter user community classification task.
- Section 6.3 introduces our proposed TBNB approach.
- Section 6.4 evaluates our proposed TBNB approach using the two ground-truth datasets generated using the two proposed ground-truth generation approaches.
- Section 6.5 provides concluding remarks for this chapter.

6.2 Automatic Ground-Truth Generation Approaches

In this section, we introduce a hashtag labelling approach in Section 6.2.1 and a DBpedia labelling approach in Section 6.2.2. These two labelling approaches generate ground-truth datasets, which are used later to train our Twitter user community classifiers.

6.2.1 The Hashtag Labelling Approach

In this section, we propose a hashtag labelling approach to generate ground-truth data for classifying communities with different political orientations during a political event, i.e. the 'Yes' community in the Scottish Independence Referendum (IndyRef, hereafter) 2014.

Hashtags, such as #YesScot or #NoThanks could be good indicators to label Twitter users supporting the 'Yes'/'No' communities in IndyRef. For example, Twitter users who support the independence of Scotland are likely to include #YesScot and #VoteYes in their tweets (e.g. in Macdowall, 2014). Therefore, we argue that hashtags can be used to generate ground-truth data for classifying Twitter users into communities in an election or a referendum. In Section 6.2.1.1, we first present how to use hashtags to generate the ground-truth data. We then introduce a method for using the Twitter followee network to validate the ground-truth data generated by using our hashtag labelling approach in Section 6.2.1.2.

6.2.1.1 Hashtag Labelling Approach for IndyRef

We describe the proposed hashtag labelling approach using IndyRef as an example. There are two communities during Indyref: the "Yes" community (in favour of independence) and the "No" community (opposed). A corpus pertaining to IndyRef was first collected from Twitter by searching for a number of referendum-specific hashtags (e.g. #IndyRef) and associated terms (e.g. 'vote', 'referendum') using the Twitter Streaming API[2]. We obtain a 33GB (uncompressed) dataset containing 6 million tweets from over 1 million unique users collected from August 1, 2014 to September 30, 2014. The most commonly used hashtags indicating the support of the two communities are listed in Sets 1 and 2 below:

- Set 1: #NoBecause, #BetterTogther, #VoteNo, #NoThanks
- Set 2: #YesBecause, #YesScotland, #YesScot, #VoteYes

As can be seen, hashtags in Set 1 were associated with a "Yes" vote, and those in Set 2 with a "No" vote. To reduce sparsity, we retain only users with more than 30 tweets posted during the time-frame of the collection. To generate our ground truth, i.e. groups of Twitter users labelled by 'No' and 'Yes', we assume that if a user's tweets are only tagged by hashtags in Set 1, then this user is labelled as a supporter in the "No" community. Similarly, if a user's tweets contain only hashtags in Set 2, then the user is labelled into the "Yes" community.

Using this method, we obtain 5326 "Yes" users and 2011 "No" users. Together these 7337 users[3] account for more than $420k$[4] tweets. After labelling, all hashtags in Sets 1 and 2 are removed from their original tweet text. The resulting tweets constitute our classification dataset (i.e. the 7337 users and their corresponding $420k$ tweets without the Sets 1 and 2 hashtags). Without these hashtags, the classification task is naturally more challenging, but

[2] `https://dev.twitter.com/`

[3] We obtain these users from Twitter data posted in two months. The number of users can be increased when the volume of tweets is increased.

[4] k, following a figure, denotes "thousand" as per standard unit. Otherwise, it means the index of a topic.

importantly, the resulting generalisable classifier does not require the presence of hashtags. Before we use this dataset (i.e. 5326 "Yes" users and 2011 "No" users) to train various classifiers, we verify the quality of this dataset using the Twitter users' followee network.

6.2.1.2 Verification of the Hashtag Labelling using the Followee Network

We verify the reliability of our hashtag labelling approach using the users' followee networks. In particular, members of the Conservative Party (CONV) were staunchly opposed to the Scottish independence, with post-election surveys showing that 95% of Conservatives voted "No" (Ashcroft, 2014). Thus, we argue that if a user mainly follows Conservative politicians, this person is likely to be a "No" voter. In contrast, 86% of the Scottish National Party (SNP) voters favoured independence(Ashcroft, 2014), and hence if a user follows SNP politicians, their vote intention is more likely to be "Yes". We then examined the networks of the 7337 users in our dataset, and used the Twitter REST API[5] to identify who these users follow among the 536 public Twitter accounts corresponding to Members of the British (MPs) or Scottish (MSPs) Parliaments. We use two verification approaches, denoted c_{V1} and c_{V2} for verifying the reliability of our ground truth: c_{V1} assumes an exclusive followee membership, while c_{V2} assumes a marked tendency (20[6] more followed politicians from one party than the other) to follow politicians of a given political party, namely:

$$c_{V1}(u) = \begin{cases} \text{“}Yes\text{”} & \text{if } n_{CONV}(u) = 0 \ \wedge \ n_{SNP}(u) > 0 \\ \text{“}No\text{”} & \text{if } n_{CONV}(u) > 0 \ \wedge \ n_{SNP}(u) = 0 \end{cases}$$

$$c_{V2}(u) = \begin{cases} \text{“}Yes\text{”} & \text{if } n_{SNP}(u) - n_{CONV}(u) > 20 \\ \text{“}No\text{”} & \text{if } n_{CONV}(u) - n_{SNP}(u) > 20 \end{cases}$$

where $n_p(u)$ is the number of times user u follows a politician (MPs/MSPs) of party p. We validate our ground truth by comparing a user's label allocated using the hashtag labelling approach versus that allocated using the two verification approaches. If the two labels are concordant, then the user voting intention is said to be verified, i.e. it is likely to be correct.

Table 6.1 reports the agreement statistics between our hashtag labelling approach and the two verification methods. We find that c_{V1} verifies more users than c_{V2}, but shows lower

[5] `https://dev.twitter.com`

[6] Considering that the total number of the used Twitter accounts of MPs/MSPs is 536, following 20 more MPs/MSPs from a given party (either SNP or CONV) than the other is a reasonably good indication of the orientation of a Twitter user. We also checked a small group of users labelled by this method and found that this method works well.

Table 6.1: Agreement between the hashtag labelling approach and our followee network verification method.

	Verified Users	Agreement Number	Agreement	Cohen's $kappa$
c_{V1}	6339	5424	0.856	0.662
c_{V2}	684	619	0.905	0.800
$c_{V2} \cup c_{V2}$	6632	5770	0.870	0.718

agreement (c.f. Cohen's $kappa$) than c_{V2}. Overall, we find that, of the 6332 supporters verified by c_{V1} or c_{V2}, 87% can be verified into "Yes" or "No" communities, demonstrating that our ground-truth produced by the hashtag labelling approach is reasonable and reliable. Later, in Section 6.4, we further evaluate our hashtag labelling approach by examining whether the generated IndyRef dataset can be used to effectively train a community classifier.

Although the user followee network can be used to label Twitter users' community affiliations, it cannot be generally applied to generate ground-truth data since not all Twitter users follow politicians on Twitter. On the other hand, it can be time-consuming to obtain users' followee networks since the Twitter REST API is required to obtain users' followees and such API has limits. Therefore, we do not use the user followee network to generate ground-truth data in this thesis. Theoretically, one can use both our hashtag labelling approach and the user followee network to generate ground-truth data. In the next section, we introduce another ground-truth generation approach, the DBpedia labelling approach.

6.2.2 The DBpedia Labelling Approach

As discussed in Section 2.5.2 and Section 3.4.3, social scientists are interested in understanding the connections among different communities, such as the media and politician communities. Therefore, it is necessary to develop a community classifier to identify communities in terms of the professions of users. As a first step, it is important to generate ground-truth data to train such a user community classifier. The existing automatic ground-truth generation approaches have limitations when generating ground-truth data for communities (see Section 3.4.2). Hence, we propose a DBpedia labelling approach to generate ground-truth labels for classifying Twitter users into communities. Next, we first describe the definitions of communities in Section 6.2.2.1. We then introduce our DBpedia labelling approach in Section 6.2.2.2. We describe two baseline approaches in Section 6.2.2.3. Section 6.2.2.4 describes the three generated datasets by using our DBpedia labelling approach and the two baseline labelling approaches. Finally, Section 6.2.2.5 describes how we conduct a user study to obtain human judgements of Twitter users' community labels, which are used to evaluate our DBpedia labelling approach.

6.2.2.1 Definitions of Communities

In this section, we aim to use our DBpedia labelling approach to identify four communities: academics, community media, business elites and politics. We choose these four communities because social scientists are interested in these four communities during an election, as mentioned in Section 1.1 and Section 3.4.3. Moreover, these four communities are major components of society. We introduce the definitions of these four communities:

- **Academics** (`ACA`): Twitter users who are doing research or teaching in academic institutes belong to Academics. Users in this group may work in a specific research field, e.g. Chemistry, Mathematics or Social Science. For example, a Twitter user who describes himself/herself as a researcher or who has an academic title (e.g. Professor, Lecturer, etc.) is an academic user. Twitter users in this community commonly post tweets about research topics, new studies, findings, papers, etc.
- **Media** (`MDA`): People who work in newspapers or broadcast companies belong to this class of users. They can be journalists, reporters, correspondents, etc. Most of them hold a neutral position when reporting some events/news. Media people usually share stories, break news or reports of trending events.
- **Business Elites** (`BE`): People from commercial companies are categorised as business users. Usually, a business elite posts tweets about their new products and the business plans of their companies or how to manage a project/company/team, etc. They may be highly likely to interact with the other business people in the relevant fields on Twitter.
- **Politics** (`PLT`): People who actively engage in discussions related to political topics are considered as `PLT`. Even people who do not have any political affiliations are considered as `PLT` if they frequently post tweets related to politics. As such, in this work, we categorise both politicians and people who are actively involved in politics into one community called `PLT`.

A Twitter user might belong to multiple communities. For example, a journalist who is interested in politics might often report political stories. To simplify the work, we assume that a Twitter user only belongs to a single community. This assumption is reasonable considering that the population of users belonging to multiple communities is not large, and hence it is likely that most of the Twitter users only belong to one of the aforementioned communities. In the next section, we explain how to use our DBpedia labelling approach to obtain labels of the four communities.

Table 6.2: Examples of combinations of DBpedia predicates & objects and the extracted DBpedia entities.

Community	Combination Predicate & Object		Extracted Entities	
	#	Examples	#	Examples
Academics (ACA)	13	Subject:Category&Science_occupations 22-rdf-syntax-ns#type&University 22-rdf-syntax-ns#type&Institution ...&...	167k	University_of_Cambridge Professor/Lecture Carl_Schmidt,chemist ...
Media (MDA)	17	Subject:Category&Journalists 22-rdf-syntax-ns#type&Broadcaster 22-rdf-syntax-ns#type&Newspaper ...&...	56k	Piers_Morgan/Fiona_Bruce National_Observer_(UK) BBC_World_News ...
Business Elites (BE)	8	Subject:Category&Business_occupations 22-rdf-syntax-ns#type&Company108058098 22-rdf-syntax-ns#type&BusinessPerson ...&...	83k	Chief_Executive_Officer Apple_Inc./HSBC Mark_Zuckerberg ...
Politics (PLT)	15	Subject:Category& Legislators Subject:Category&Political_occupations 22-rdf-syntax-ns#type&PoliticalParty ...&...	93k	Theresa_May President/Major Conservative_Party_(UK) ...

6.2.2.2 The Implementation of the DBpedia Labelling Approach

Our approach mimics the ways in which humans would distinguish a community, i.e. identify whether a Twitter user uses the community-related keywords (e.g. "professor" for academics and "journalist" for media) in their tweets. Therefore, as a first step, it is important to build on the prior knowledge of the community-related keywords. However, it is challenging to obtain these community-related keywords. DBpedia is a widely used knowledge base and it contains many links to other knowledge bases (Mendes et al., 2011). The entities in DBpedia are well structured (i.e. in n-triple format) (Färber et al., 2015), which allows us to easily select community-related entities from the DBpedia knowledge base. Hence, we choose to use entities from DBpedia[7] as the community-related keywords. An entity in DBpedia is usually the name of the Wikipedia article, e.g. the name of a person or an organisation. We argue that the DBpedia entities that are used in Twitter users' tweets can indicate their community affiliations. Once we obtain the community-related DBpedia entities for each of the four communities, we can assign Twitter users to a community according to their usage of these community-related DBpedia entities. In the following, we describe the two steps of our proposed DBpedia labelling approach:

Step 1: Extracting community-related entities. In the DBpedia knowledge base, each entity is represented in an n-triple format in a Resource Description Framework[8] (Bizer et al., 2009). Such a format is <subject> & <predicate> & <object>, where

[7]We acknowledge that some other knowledge bases can also be used, such as Freebase (https://developers.google.com/freebase/).

[8]https://www.w3.org/RDF/

<subject> is the entity and n combinations of <predicate> & <object> are used to describe this entity. Usually, the n combinations of <predicate> & <object> describe the properties of an entity. We list 3 combinations of <predicate> & <object> describing the entity "Professor":

Professor &**Subject** & **Academic_terminology**
Professor &**Subject** & **University_and_college_people**
Professor &22-rdf-syntax-ns#type & Ting
...

By distinguishing <predicate> & <object>, we can identify that the entity "Professor" belongs to the community ACA since Subject & Academic_terminology and Subject & University_and_college_people are academic-related. Therefore, we first manually select the combination rules of <predicate> & <object> for our chosen four communities. We list several examples of these combinations in Table 6.2[9]. Second, in order to obtain enough entities, we extract entities from four commonly used DBpedia knowledge bases: instance types, article categories, YAGO types, UMBEL link[10].

For each entity in these 4 DBpedia knowledge bases, if it can be described by the pre-set <predicate> & <object> combinations of community c, i.e. the pre-set combinations of community c is contained in the n-triple formatted entries of entities, we say that these entities are related to community c ($c \in \{$ACA, MDA, BE, PLT$\}$). We ignore the entities that can be described by the <predicate> & <object> combinations of different communities, since these entities could generate Twitter users which might belong to different communities. Finally, we obtain a certain number of entities for the four communities listed in Table 6.2 (column "Exacted Entities"), e.g. the community PLT has $93k$ entities containing the names of important politicians, political parties, and people actively involved in politics. Next, we explain how we use these extracted community-related DBpedia entities to obtain the ground-truth Twitter users from a collection of tweets.

Step 2: Twitter users Filtering. We examine whether both the profile description and the recent tweets of each of the candidate Twitter users[11] have used the community-related DBpedia entities (extracted using the first step) and then label the Twitter user accordingly. In our DBpedia labelling approach, we assume that a Twitter user has only one community

[9]The full list of the combinations Predicate & Object for the four communities are listed in Tables A.2 and A.3 in the Appendix.

[10]These 4 datasets are in n-triple format and can be downloaded from https://wiki.dbpedia.org/downloads-2016-10. Among these 4 datasets, instance types and article categories are two datasets of DBpedia while YAGO types and UMBEL link are two external datasets that link to DBpedia.

[11]These users can be obtained from a collection of tweets, which can be crawled using the Twitter Streaming API (https://dev.twitter.com). Note that this data collection method is the same as the collection method in Section 6.2.1.

Table 6.3: The Twitter public lists used in the baseline labelling approaches for generating ground-truth data.

Community	Number of lists	Twitter public lists
ACA	8	Higher Ed Thought Leaders(@MSCollegeOpp), Edu-Scholars(@sesp_nu), Favourite academics(@AcademiaObscura), Northwestern(@sesp_nu), SESP Alumni(@sesp_nu), STEM Academic Tweeters(@LSEImpactBlog), The Academy(@AcademicsSay), Harvard(@hkslibrary)
MDA	16	Mirror Political Journos(@MirrorPolitics), Mirror reporters/columist(@DailyMirror), sundaymirror(@DailyMirror), Financial Tweets(@TIME), TIME Staff(@TIME), Sun accounts(@TheSun), Sun people(@TheSun), BBC News Official(@BBCNews), BBC Asian Network(@BBC), BBC News(@BBC), Business staff(@guardian), Observer staff(@guardian), Money staff(@guardian),Technology staff (@guardian), Politics staff(@guardian), News staff (@guardian)
BE	5	Social CEOs on Twitter(@debweinstein), Tech Startup Founders(@realtimetouch), Top CEO's(@chrisgeorge187), Tech, Startups & Biz(@crblev), Awesome Entrepreneurs(@vincentdignan)
PLT	6	UK MPs(@TwitterGov), US Governors(@TwitterGov), US Senate(@TwitterGov), US House(@TwitterGov), Senators(@CSPAN), New Members of Congress(@CSPAN)

affiliation. If a Twitter user's description and more than 20%[12] of the user's recently posted tweets use the entities from the same community c, then this user is labelled as a member of community c in our dataset. Otherwise, the user is excluded. Instead of directly checking whether the text of a tweet/profile contains the community-related entities, we use DBpedia Spotlight (Daiber et al., 2013) to identify the DBpedia entities in the users' profile descriptions and tweets.

Note that the DBpedia labelling approach cannot be directly used as a classifier. The approach can only label Twitter users when these users mention DBpedia entities. However, many profile descriptions and tweets do not refer to the DBpedia entities. This means that the DBpedia labelling approach can have a very low recall. Thus, it cannot generally be used to classify the community affiliations of Twitter users. To generally classify Twitter users into communities, it is necessary to build a classifier using the ground-truth dataset generated by the DBpedia approach. In the following sections, we introduce two baseline labelling approaches (Section 6.2.2.3), which are compared to our DBpedia labelling approach. Section 6.2.2.4 describes the three datasets generated using our DBpedia labelling approach and the two baseline labelling approaches, respectively. To study the quality of the ground-truth data generated using our DBpedia labelling approach, we conduct a user study introduced in Section 6.2.2.5.

6.2.2.3 Two Baseline Labelling Approaches

As discussed in Section 3.4.2, the uses of emotion symbols and pre-defined POS patterns cannot label user into communities automatically. However, one possible method to generate

[12] In our preliminary experiments, we applied the DBpedia labelling approach on a small size of Twitter data. We found that 20% was a sufficient proportion of tweets to indicate their community affiliation. Considering that not all tweets have entities, a higher proportion can reduce the size of the ground-truth data while a lower proportion may introduce more noises. Hence, we use 20% as the threshold in this work.

ground-truth data is to use the Twitter public lists as proposed by Su et al. (2018). Such lists often contain a group of Twitter user accounts that belong to the same community. For example, a Twitter user `@sesp_nu` created a public list `Edu-Scholars` (see Table 6.3), and described the list as "a selection of the nation's most influential academics in education". All of the Twitter users in this list can be then simply labelled to belong to `ACA`. Similarly, the public list `Awesome Entrepreneurs` covers a collection of entrepreneurs, which can be used to construct the `BE` (Business Elites) community. Therefore, ground-truth data can be generated using these public lists. We use two derived baseline labelling approaches:

1) The baseline approach.

We use the Twitter public lists shown in Table 6.3 (same as Su et al. 2018) to label the Twitter users into four communities, i.e. we use 8, 16, 5 and 6 existing public lists for `ACA`, `MDA`, `BE` and `PLT`, respectively. For example, the Twitter users in the public list `UK MPs` are labelled to belong to the `PLT` community. Note that all the used lists are collected manually. It can be time-consuming to understand the meaning of these lists and link them with a specific community. Hence, it can be challenging to apply this labelling approach.

2) The refined baseline approach.

It is often difficult to evaluate the credibility of pre-defined Twitter lists, which may categorise Twitter user accounts into the wrong communities. Su et al. (2018) worked on refining/cleaning the Twitter users from the Twitter public lists in order to obtain a better classification performance. They aimed to manually remove several categories of noisy Twitter users (i.e. Twitter users that cannot informatively represent the given community label). For instance, one noisy category is that of users who have clear community affiliations (indicated by their involved Twitter public lists), but who often posted tweets that do not relate to their declared communities and hence are likely to belong to other communities. Su et al. (2018) reported that the classification performance can be improved after the Twitter users in this category were removed. Therefore, we remove Twitter users in this noisy category as our refined baseline approach. Next, we describe the three generated ground-truth datasets using the DBpedia labelling approach and the two baseline labelling approaches.

6.2.2.4 The Three Generated Training Datasets

We describe the three training datasets generated using our DBpedia approach (c.f. Section 6.2.2.2) and the two baseline approaches (c.f. Section 6.2.2.3). These training datasets are used later to train the Twitter user community classifiers.

Table 6.4: The number of the labelled users of four communities generated using our DBpedia labelling approach.

	ACA	MDA	BE	PLT
users	4982	5512	$22k$	4538
tweets	$84k$	$97k$	$430k$	$106k$

1) DBpedia Training Dataset.

We use a collection of tweets, called the Twitter background dataset, which contains 540 million tweets (10% tweet samples collected using the Twitter Streaming API) posted from September 2015 to March 2016. First, Twitter users who posted less than 15 tweets over the 7 months period are removed from this Twitter background dataset, since these Twitter users are inactive. After applying our DBpedia labelling approach (c.f. Section 6.2.2.2) on this Twitter background dataset, we obtain a collection of Twitter users with community labels, as shown in Table 6.4, which is larger than most of the training datasets for Twitter user classification in the literature. For example, to classify Twitter users by their genders, Al Zamal et al. (2012) used about 400 Twitter users with known genders for training a gender classifier. Conover et al. (2011b) predicted the political alignment of Twitter users using approximately 900 ground-truth users as training data. It is worth mentioning that the size of this dataset can be easily enlarged by increasing the size of the Twitter background dataset. For all ofthe labelled Twitter users, we extract their tweets from the Twitter background dataset and filter out non-English[13] users/tweets. To verify the quality of this generated ground-truth data, we randomly select 200 users from each community (800 in total) for conducting our user study (further described in the next section). In our classification experiments, we select approximately $5k$ users (see "DBpedia training data" row in Table 6.5) from each of the communities as the training dataset. We call this the DBpedia training dataset.

2) Baseline and Refined Baseline Training Datasets.

As explained in Section 6.2.2.3, we select Twitter users from existing Twitter public lists as a training dataset. For each community, we crawl[14] roughly 1000 users with their 20 recent tweets. This forms our **baseline training dataset**. Next, all Twitter users in the described noisy category (see Section 6.2.2.3) are removed from the baseline training dataset, which is denoted the **refined baseline training dataset**. The sizes of the datasets across the four communities are listed in Table 6.5.

[13]The language used in a tweet is identifiable, see `https://developer.twitter.com/en/docs/developer-utilities/supported-languages/api-reference/get-help-languages.html`.

[14]Using the Twitter REST API (`https://dev.twitter.com`).

Since these three ground-truth generation approaches are different and generate a dataset containing different users, we cannot directly compare them. Therefore, we train user community classifiers using the three generated ground-truth datasets shown in Table 6.5. We evaluate the effectiveness of the three approaches by analysing the performance of these trained classifiers using an identical test data. Measuring the classification performance allows us to determine whether these ground-truth generation approaches are suitable for user community classification. More details are reported in Section 6.4.5. To obtain the test data, we conduct a user study to obtain human judgements of the community labels of Twitter users in the next section. At the same time, this user study also allows to validate the ground-truth data generated using our DBpedia labelling approach.

Table 6.5: The number of users in the training dataset for the DBpedia community classification task.

	ACA	MDA	BE	PLT
DBpedia training dataset	4782	5312	4800	4339
Baseline training dataset	812[15]	1000	1000	1000
Refined baseline training dataset	590	590	589	590

6.2.2.5 User Study

The quality of the ground-truth datasets generated by using the two baseline labelling approaches (c.f. Section 6.2.2.3) have been verified manually in Su et al. (2018). In this section, we conduct a user study to identify the quality of the ground-truth dataset generated using our DBpedia labelling approach. We first randomly sample 200 Twitter users from each of the four communities from the ground-truth data generated using our DBpedia labelling approach. We present these Twitter users to crowdsourcing workers in order to obtain the human ground-truth labels of the 800 Twitter users. Similar to Chapter 4, we use the CrowdFlower platform as a source of workers. Next, we first describe our user study, followed by the obtained results.

Description of the user study. We present the crowd-sourcing workers with the user profile of a given Twitter user and their 8[16] recent tweets. The 8 recent tweets can assist a worker to make a decision on whether a Twitter user belongs to a community. After reading both the Twitter user's profile description and tweets, the worker is asked to choose one community label from the community labels by considering the definitions of four communities

[15]Users are removed from the 800 if their user profiles cannot be accessed.

[16] More tweets can make the user study task difficult for workers. We choose the 8 recent tweets of Twitter users since 8 tweets can reasonably represent a Twitter user and a crowdsourcing worker can read and understand 8 tweets in a short period of time.

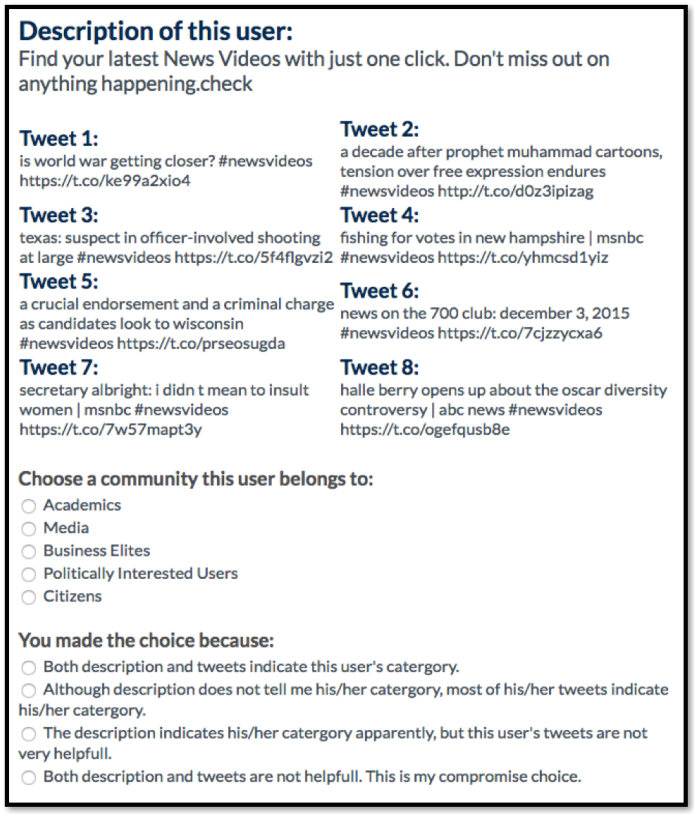

Figure 6.1: The user interface for obtaining the human judgements of community labels.

described earlier in Section 6.2.2.1. It is possible that the Twitter user does not belong to any of the four communities. Hence, we introduce another label "Citizen" in the user study, which encapsulates all other possible communities, i.e. a catch-all. Indeed, we instruct a crowdsourcing worker to label a Twitter user who apparently does not belong to the four communities as a citizen, i.e. belonging to other communities. The user interface for this assessment task is shown in Figure 6.1. A worker is paid $0.05 for each judgement, and we obtained 3 independent judgements for each Twitter user. To control the quality of this user study, we first give workers a set of test assessments, where the responses (the community label of a Twitter user) are verified in advance. CrowdFlower workers can enter the task only if 70% of their answers for our test items are correct.

Results of the user study. 800[17] chosen Twitter users from our ground-truth (generated by using the DBpedia labelling approach) are evaluated by 124 unique English-speaking Crowdflower workers. Among them, 97 workers are from the US and the rest are either from the UK or from Canada. In total, we obtained 4,868 assessments, with each worker contributing about 39.3 assessments on average. Each Twitter user is assigned a label based on the majority vote from the Crowdflower workers. It is worth noting that 92.7% of Twitter

[17]We choose 200 Twitter users for each of the four communities.

users received at least 2 consistent categorisations from 3 different human workers, which indicates that at least 2 workers have an agreement in most circumstances. Instances where there were three different labels occurred only 7.3% of the time. For these 7.3% Twitter users, more judgements were made until a majority was reached.

Table 6.6: The comparison between human judgements and the DBpedia labelling approach.

	ACA	MDA	BE	PLT	Citizen
Accuracy	0.558				
Cohen's $kappa$	0.444				
F1	0.631	0.534	0.638	0.673	0.0
Precision	0.485	0.560	0.675	0.512	0.0
Recall	0.906	0.511	0.605	0.980	0.0

Table 6.7: Confusion matrix of the users' community labels between the DBpedia labelling approach and human assessors.

		DBpedia labelling approach			
		ACA	MDA	BE	PLT
Humans	ACA	97	4	4	2
	MDA	43	112	30	34
	BE	27	47	135	14
	PLT	2	0	0	102
	Citizen	31	37	31	47

When we compare the labels of the 800 Twitter users judged by the Crowdflower workers to those from our DBpedia labelling approach, we observe that the accuracy of our DBpedia labelling approach is 0.558 and the $kappa$ agreement (Cohen's $kappa$ agreement (Artstein and Poesio, 2008)) is 0.444, as shown in Table 6.6. According to Fleiss et al. (2003), this agreement suggests that our DBpedia labelling approach has a good agreement with human judgements considering that the random probability of a user being labelled into any category is 20%. An accuracy of 0.558 suggests that there is noise in the ground-truth data generated by our DBpedia labelling approach. However, this does not mean that this generated ground-truth data cannot be used in our community classification task. In Section 6.4.5, we report the performance of classifiers trained using this ground-truth data on a test data, i.e. the 800 Twitter users with human-annotated community labels.

Table 6.6 also reports the Precision, Recall and F1 scores of the DBpedia labelling approach compared to the ground-truth obtained from human assessors. `ACA` and `PLT` obtain high recall but low precision, which indicates that the DBpedia labelling approach tends to categorise the users into these two communities more than the others. In the confusion matrix shown in Table 6.7, we observe that the `BE` Twitter users can be better labelled by the DBpedia labelling approach since the number of true positive of `BE` is 135, which is higher

Figure 6.2: Example of a Twitter user.

than the other 3 communities. The Twitter users in BE can be misclassified as MDA. The reason for this could be that our DBpedia labelling approach has difficulties when identifying users who might belong to multiple communities. To illustrate, we present a Twitter user shown in Figure 6.2, where the word "Mumpreneur" is most likely an indicator of a business person. However, in the content of this user's tweets, there are news/articles about business, social media and technology which might be indicative of other communities. In addition, a number of Twitter users are labelled as "Citizens" (i.e. belonging to other communities) in our user study. It might because these Twitter users do not belong to one of the four communities or the given 8 tweets do not clearly indicate their community affiliations. However, most of the Twitter users in the four communities can be labelled correctly by our DBpedia labelling approach.

So far, we have validated the ground-truth data generated using the DBpedia labelling approach. We also obtain a test data for the DBpedia community classification, i.e. 97 ACA users, 112 MDA users, 135 BE users and 102 PLT users as shown in Table 6.7.

6.2.3 Summary

We have introduced two ground-truth labelling approaches: the hashtag labelling and the DBpedia labelling approaches. Using these two approaches, we have obtained two ground-truth datasets for training user community classifiers on two Twitter datasets, i.e. the IndyRef dataset and the DBpedia community dataset. These datasets are used to conduct our Twitter user community classification experiments. Next, we introduce a new approach to classify Twitter users into communities.

6.3 Topic-Based Naive Bayes — TBNB

To effectively identify the community labels of Twitter users, we propose a Topic-Based Naive Bayes approach, namely TBNB. We first introduce our TBNB approach by analysing the topics discussed during IndyRef (Section 6.3.1). Then, we introduce how we implement our TBNB approach (Section 6.3.2).

6.3.1 Topics Analysis in IndyRef

The IndyRef discussions on Twitter revolved around a number of topics, for which people's opinions usually reflected their vote intentions. For example, many "Yes" voters believed that revenues derived from the North Sea oil fields belonged to Scotland and could sustain its economy. On the other hand, many "No" voters argued that these sources were insufficient in the long run. For the same topic, two communities discussed them differently, i.e. use different words. A word is used as a feature in a classifier. A feature's *dissimilarity* represents the usage difference of this feature across topics. For example, the difference in usage of "oil" across different topics is high. For a given topic, a feature's *variance* refers to the difference of the conditional probabilities of the occurrence of such a feature in different communities. For example, the conditional probability of "oil" in the "Yes" community is higher than in the "No" community. Typically, the feature selection approaches select features with higher variances between communities. Thus if a feature differs between topics (i.e. its dissimilarity is high), it will be treated as different features in our TBNB classifier. Thus TBNB can capture term dependencies between topics and user voting intentions. On the other hand, since the essence of the Naive Bayes (NB) classifier is to learn those features with high variance from the communities, we argue that the TBNB classifier can work better by leveraging both the features' dissimilarities across topics and their variances in the communities.

6.3.2 Implementation of TBNB

We assume that a single tweet involves a single topic since a tweet is short. In the training step, we first apply LDA (c.f. Section 2.2.1) to extract K topics from the training dataset:

$$\{topic_1, ..., topic_k, ..., topic_K\} \tag{6.1}$$

where K is the total number of topics. For each topic k, a corresponding probability table is produced, i.e. $p(w|c, topic_k)$ and $w \in V$[18], where each feature (word w) has two associated

[18] V is the word vocabulary.

conditional probabilities related to the two possible voting intentions (i.e. $C = \{Yes, No\}$):

$$p(w|c, topic_k) = \frac{number\ of\ the\ word\ w\ in\ topic_k\ in\ c}{total\ number\ of\ words\ in\ topic_k\ in\ c},\ c \in C,\ k \in \{1, .., K\} \quad (6.2)$$

Consequently, during the training step, we produce as many feature tables as the number of used topics (i.e. K). In the testing step, we treat each user as a virtual document where the virtual document contains the users' tweets, i.e. $user = \{tw_1, ..., tw_j..., tw_J\}$, where J is the total number of tweets a user has. For each tweet tw, we can first obtain its topic distribution θ^{tw} using Equations (2.6) and (2.7) (described in Section 2.2.2). Then we can associate tw with its closest topic by using:

$$closest(tw) = \underset{k \in \{1,...,k\}}{arg\ max}\ (\theta_k^{tw}) \quad (6.3)$$

where θ^{tw} is the topic distribution of the tweet tw. The closest topic is selected when the conditional probability of a topic on this tweet is highest[19]. Next, terms in an unseen tweet are then examined using the probability table generated during the training step for the topic with which this tweet is associated. The community affiliation of a Twitter user can be computed as follows:

$$\begin{aligned} community(user) &= \underset{c \in C}{arg\ max}(p(c|\boldsymbol{W}_{user}, \{topic_1, ..., topic_K\})) \\ &\propto \underset{c \in C}{arg\ max}(p(c) \times p(\boldsymbol{W}_{user}, \{topic_1, ..., topic_K\}|c)) \\ &= \underset{c \in C}{arg\ max}(p(c) \times \prod_{w \in \boldsymbol{W}_{user}} p(w|c, topic_{k'}) \times p(topic_{k'}|c)) \end{aligned} \quad (6.4)$$

$$p(topic_{k'}|c) = \frac{number\ of\ tweets\ belonging\ to\ topic_{k'}\ in\ c}{number\ of\ tweets\ in\ c} \quad (6.5)$$

$$p(c) = \frac{number\ of\ users\ belonging\ to\ c}{total\ number\ of\ users} \quad (6.6)$$

where $\boldsymbol{W}_{user}$ is a group of words in a user and k' is the index of the closest topics of each word w calculated using Equations (6.3)[20]. In this way, terms in different tweets are treated differently based on their most closely associated topics, and the TNBN classifier applies, for each unseen tweet, those features that were learned from the corresponding topics. We list the algorithm of TBNB in Algorithm 2, where Equations (6.2) and (6.6) are used in the training step and Equation (6.4) is used in the test step to identify the community labels of Twitter users.

[19] Here we assume that a tweet only discusses one topic to simplify TBNB.

[20] All the words in a tweet are assigned to the topic index k' if topic k' is closest to the tweet.

Algorithm 2: Topics-Based Naive Bayes (TBNB).

$topic_k, k = \{1, 2, .., K\} \leftarrow topic\_detection(tweets_{training})$
$k \leftarrow 1,\ C = \{\text{“}Yes\text{”}, \text{“}No\text{”}\}$
Training:
for $k \leq K, k++$ **do**
  if w **in** $topic_k$ **then**
    $p(w|c, topic_k) = \frac{number\ of\ the\ word\ w\ in\ topic_k\ in\ c}{total\ number\ of\ words\ in\ topic_k\ in\ c}, \forall c \in C$ (i.e. Equation (6.2))
  end if
end for
$p(topic_{k'}|c) = \frac{number\ of\ tweets\ belonging\ to\ topic_{k'}\ in\ c}{number\ of\ tweets\ in\ c}$ (i.e. Equation (6.5))
$p(c_i) = \frac{number\ of\ users\ belonging\ to\ c_i}{total\ number\ of\ users}$ (i.e. Equation (6.6))
Testing:
$community(user) = arg\ max_c(p(c) \times \prod_{w \in \boldsymbol{W}_{user}} p(w|c, topic_{k'}) \times p(topic_{k'}|c)),$
$where\ w \in \boldsymbol{W}_{user},\ c \in C$ (i.e. Equation (6.4))

We have introduced our TBNB classification approach in this section. In the next section, we evaluate our TBNB classification approach using the datasets generated using the two proposed ground-truth generation approaches (described in Section 6.2).

6.4 Evaluation

In this section, we evaluate our proposed TBNB approach for community classification. At the same time, we assess whether our proposed hashtag labelling and DBpedia labelling approaches can generate reasonable datasets for the community classification task. We conduct two community classification tasks: 1) the IndyRef community classification task and 2) the DBpedia community classification task, where classifiers are trained using datasets generated using the hashtag labelling and the DBpedia labelling approaches, respectively. In particular, we aim to identify the ‘Yes’ and ‘No’ communities in the IndyRef classification and to classify Twitter users into four communities[21] in the DBpedia community classification. For both tasks, we apply our TBNB classification approach. Next, we first report the summary of the datasets generated using the two ground-truth generation approaches in Section 6.4.1. We then describe the experimental setup for the two tasks in Section 6.4.2. We list the research questions in Section 6.4.3 and analyse the results in Sections 6.4.4 and 6.4.5.

[21] In our user study (c.f. Section 6.2.2.5), we introduced the label of ‘Citizens’ (i.e. designating the other communities) to help the crowdsourcing workers conduct the task. We do not use the class of ‘Citizens’ in our DBpedia community classification task due to the irrelevant noise it contains.

6.4.1 Datasets

We use the IndyRef dataset (c.f. Section 6.2.1) generated using the proposed hashtag labelling approach for the IndyRef community classification task. For the DBpedia community classification task, we generate three datasets (c.f. Section 6.2.2.4) using the two baseline approaches and our proposed DBpedia labelling approaches. We list the details of these datasets in Table 6.8. For the IndyRef community classification, we use a 10-fold cross-validation process over the 7337 users of our dataset to evaluate the performance of our classifiers. For the DBpedia community classification, we do not perform cross-validation since the quality of the DBpedia community dataset is not as high as the IndyRef dataset (see Table 6.1 versus Table 6.6). Instead, we use the human-verified Twitter users from our user study as the test dataset. The size of this test data is indicated in row "Test dataset" of Table 6.8. All the users in the test dataset do not appear in the training data.

As our test dataset in the DBpedia community classification task, we focus our experiments on the Twitter users obtained from our DBpedia dataset, instead of those users obtained from the baseline and refined baseline datasets[22]. This is because we aim to classify the community affiliations of *general users* (users that are not significant figures, i.e. they belong to the general public) during a political event. The Twitter users in the dataset generated using our DBpedia labelling approach are randomly sampled users from the Twitter Stream[23]. However, the Twitter users in the baseline and refined baseline datasets are well-known persons in their communities and they cannot represent general users on Twitter. For example, the Twitter users in `Higher Ed Thought Leaders` are famous scholars who have a big achievement in the `ACA` community. In contrast, a general Twitter user in the ACA community could be a PhD student, an early career scholar or a less famous academic. Therefore, the Twitter users in the baseline dataset are not general enough. As a result, we use Twitter users from the dataset generated using our DBpedia labelling approach as our test dataset, i.e. the 800 Twitter users whose community affiliations are verified by the human assessors in our user study. These 800 Twitter users can be anyone who uses Twitter and they are not well-known figures as in the baseline dataset.

[22]To investigate the generalisation of our DBpedia labelling approach, we also report experiments using the Twitter users from the refined dataset (see Section 6.4.5 for more details).

[23]Morstatter et al. (2013) compared a collection of tweets from the Twitter stream (obtained using the Twitter Streaming API) to a more general collection of tweets (crawled using the Twitter Firehose, see `https://developer.twitter.com/en/docs/tweets/sample-realtime/overview/decahose.html`). They found that the topics extracted from these two collections of tweets were almost the same, which suggests that the Twitter users crawled using the Twitter Stream API are general.

Table 6.8: Two datasets for our user community classification tasks. (a) The dataset generated using our hashtag labelling approach. (b) The dataset generated using our DBpedia labelling approach and two baseline labelling approaches.

(a). IndyRef Community Classification

Datasets	Yes	No
IndyRef dataset	5326	2011

(b). DBpedia Community Classification

Datasets	ACA	MDA	BE	PLT
DBpedia training dataset	4782	5312	4800	4339
Baseline training dataset	812	1000	1000	1000
Refined baseline training dataset	590	590	589	590
Test dataset	97	112	135	102

6.4.2 Experimental Setup

In this section, we first explain the classification setup and topic setup in Sections 6.4.2.1 and 6.4.2.2, respectively. We then explain how we apply feature selection approaches in Section 6.4.2.3, followed by the used metrics described in Section 6.4.2.4.

6.4.2.1 Classification Setup

For both the IndyRef and DBpedia community classification tasks, we apply our TBNB approach together with commonly used classification approaches (implemented using `scikit-learn`[24]) as previous reviewed in Section 2.4.4, namely: Naive Bayes (multinomial Naive Bayes, NB), Decision Trees (DT), Support Vector Machine (SVM) and Multilayer Perceptron (MLP). For comparison, we deploy a random classifier (RDN), which generates classification results by considering the distribution of communities (i.e. classes) in the ground-truth data. We use words as features in our classification experiments. Several commonly used text cleaning techniques are used: removing stopwords and stemming (c.f. Section 2.4.2). Instances are all transformed into `TF-IDF` vectors (c.f. Section 2.4.3) as the input of all classifiers except NB. For NB, we apply feature selection approaches further explained in Section 6.4.2.3. We also apply a `one-vs-rest` strategy (e.g. as used by Weston et al., 1999) in SVM and TBNB for multi-class classification, which is the setting for the DBpedia community classification. For the MLP classifier, we set one hidden layer with 500 neurons[25]. The penalty parameter for SVM is set to 0.01[26]. For the rest of classifiers, we use their default settings in `scikit-learn` .

[24] `https://scikit-learn.org`

[25] In our preliminary experiments, we set different numbers of neurons and we found that the MLP classifier performed well when the number of neurons was set to 500. Hence, we use 500 neurons in our experiments.

[26] We ran a gird search on the penalty parameter and found 0.01 is the best setting for SVM.

Table 6.9: Topics and associated words in IndyRef. For each topic, the top 5 words (ranked by the conditional probabilities of words in topics) are listed in column "Associated Words".

Topic	Tweets%	Associated Words
currency	20.25%	currency, money, change, pay, future
salmond	15.88%	salmond, alex, debate, audience, answer
glasgow	10.95%	glasgow, team, games, great, gold
women	9.82%	patronisingbtlady, women, undecided
oil	7.91%	oil, sea, privatisation, billion, gas, cuts
fear	7.87%	country, future, voting, fear, change
lastnight	7.32%	tonight, undecided, time, wearenational
debt	7.03%	scottish, debt, government, share, pay
weapon	6.84%	nuclear, weapon, clyde, year, glasgow
edinburgh	6.13%	edinburgh, johnjappy, minister, time

6.4.2.2 Topic Setup in TBNB

We use LDA as implemented in Mallet[27]. We investigate various topic numbers ($K = \{5, 10, 20, 30\}$). Table 6.9 shows the topic terms extracted using LDA for 10 topics in the IndyRef dataset. For readability purposes, the first column of Table 6.9 provides the general theme of the extracted topic[28]. For example, we can see that tweets related to *currency* and *oil* were common. Other often used topics and features included references to Alex Salmond, who was both the leader of the Scottish National Party (SNP) and of the "Yes" campaign in 2014.

6.4.2.3 Feature Selection

For the IndyRef community classification task, we apply feature selection approaches to comprehensively compare our TBNB approach to the NB approach since TBNB is based on NB. As described in Section 2.4.3, the following feature selection approaches are commonly used for the NB classifier:

- **FR**: $Frequency(word)$ (c.f. Equation (2.13))
- **LR**: $LogProbRatio(word)$ (c.f. Equation (2.14))
- **ER**: $ExpProbRatio(word)$ (c.f. Equation (2.15))
- **OR**: $OddsRatio(word)$ (c.f. Equation (2.16))
- **WRO**: $WeightedOddsRatio(word)$ (c.f. Equation (2.17))
- **NO**: No feature selection is applied, i.e. all the words in the dataset are used as features.

[27] `http://mallet.cs.umass.edu`
[28] These themes are manually annotated.

Each selection approach ranks and selects the F (the number of the selected features) most informative features based on the training data. Of course, not every selected feature will appear in the unseen test tweets - we denote the number of such "activated" features as F_{test}. For instance, a testing tweet containing "Scotland has remained in the media spotlight throughout 2014" has 9 terms. If only "Scotland", "remained", "media" and "spotlight" were selected as features, the number of activated features would be $F_{test} = 4$. We vary the number of selected features F and the deployed feature selection approach for both NB and TBNB. At the same time, we vary the number of topics T in the TBNB classifier. Since the number of unique terms in our collection is 200k, we vary $F = \{5k, 10k, 20k, 50k, 100k, 120k, 150k, 180k\}$ for NB, for TBNB, as F depicts the number of features selected for each topic (i.e. the total number of features would be $F \times K$), we do not experiment with $F > 100k$[29]. On the other hand, for the DBpedia community classification task, we only apply the FR feature selection approach to compare TBNB to NB since we mainly aim to assess the DBpedia labelling approach compared to the two baseline labelling approaches, i.e. we simply use the $20k$ most frequent words as features since this setting allows a good performance in our experiments.

6.4.2.4 Metrics

Three standard classification metrics are used to evaluate the performance of a classifier for each class: Precision, Recall and F1. We also use an accuracy score to measure overall performance of classifiers over all the communities (two for Indyref and four for DBpedia). In addition, to compare our TBNB approach to the NB approach, we also use the following performance indicators:

- **Indicator 1: Average Number of Activated Features** F_{test}. For an unseen Twitter user, we concatenate their posted tweets into a virtual document and count the number of selected features activated in the virtual document. We average these numbers across the 10 folds to obtain F_{test}. Intuitively, the higher F_{test}, the greater the confidence in the predicted community since more features are used by a classifier to determine the community of a user in the test data.
- **Indicator 2: Average Rank of the Activated Feature** R_{test}. Each feature is ranked by the applied feature selection approach. This indicator represents the average rank position of all testing features of all users in the 10 folds. Intuitively, it reflects the average effectiveness level of the activated features. A higher R_{test} is better since more effective features are used by a classifier to identify the community of a user in test data.

[29] In our dataset, no topic has more than $100k$ features.

Note that we do not apply these two indicators in the DBpedia community experiments since we focus on evaluating our DBpedia labelling approach in comparison to the two baseline labelling approaches in our DBpedia community experiments.

6.4.3 Research Questions

We aim to answer the following four research questions:

- **RQ1.** Do the topic features used in TBNB improve the performance of the NB classifier?
- **RQ2.** Can our TBNB approach outperform the commonly used text classifiers?
- **RQ3.** Does the DBpedia labelling approach outperform the two baseline labelling approaches (c.f. Section 6.2.2.3)?
- **RQ4.** Do the results of TBNB obtained on the IndyRef dataset generalise to the DBpedia dataset?

6.4.4 Analysis of the IndyRef Community Classification Task

In this section, we focus on evaluating the performance of our TBNB approach. We first compare our TBNB approach to the classical NB approach. Then, we evaluate the TBNB approach compared to the other commonly used classifiers.

Figures 6.3 (a)-(e) show the performance of the NB and TBNB classifiers when varying F and K. Both NB and TBNB perform poorly when F is low. However, TBNB classifiers markedly outperform NB_NO (the NB classifier using all words as features, i.e. without using any feature selection approach) when F ranges from $10k$ to $50k$. The highest accuracy of TBNB (90.4%) is achieved when applying the WOR feature selection approach (TBNB_WOR) with K=10 and when the FR feature selection approach is deployed (TBNB_FR) with K= 5. This is a 7.8% absolute improvement over NB_NO (82.6%). When varying the number of used topics (K), we note that the performance of the TBNB classifier generally increases as K increases. However, once K reaches 30 topics (see Figure 6.3 (d)), the accuracy of TBNB starts decreasing while still outperforming NB_NO. On the other hand, each NB or TBNB classifier with feature selection approaches has an optimal F. For instance, the optimal F of NB_OR is 150k while that of TBNB_FR is 5k. Among these feature selection approaches, we find that FR is more stable than the others. For example, in Figures 6.3 (a)-(e), the red lines do not change as much as the other lines for the two classifiers.

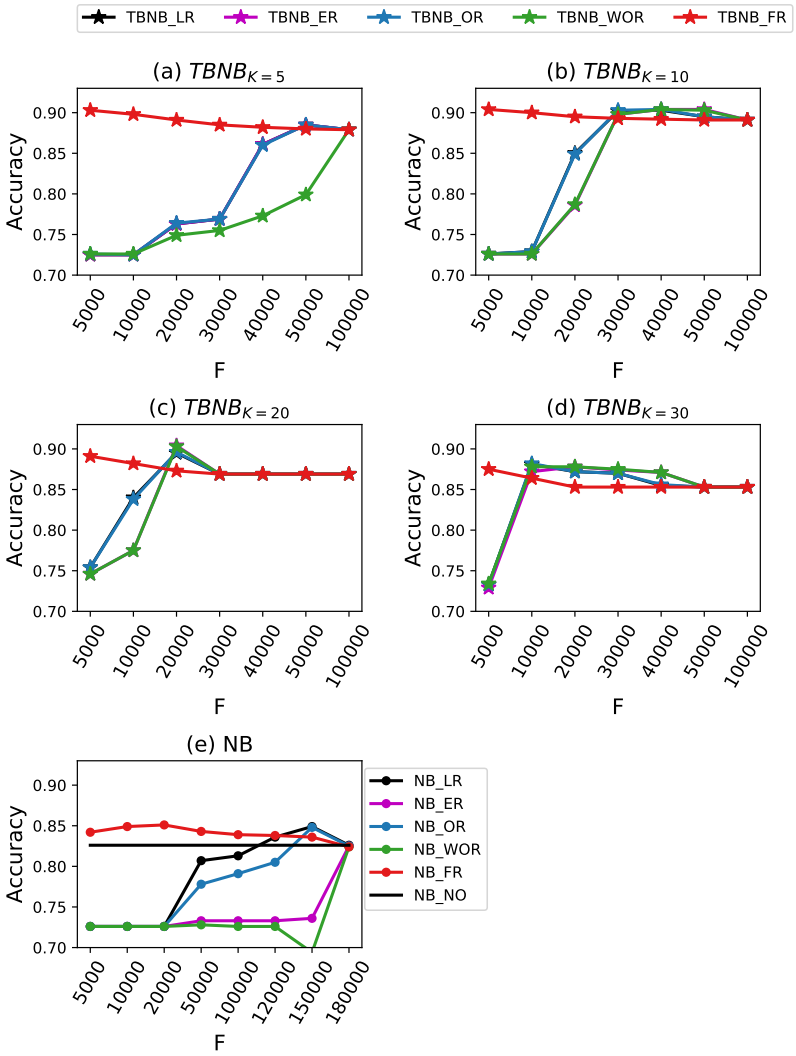

Figure 6.3: The results of NB and TBNB in the IndyRef community classification task. (a), (b), (c) and (d) show the accuracy of TBNB where K is set to 5, 10, 20 and 30, respectively; (e) The accuracy of NB. In (a), the blue line overlaps with the black and purple lines. In (b) and (c), the blue lines overlap with the black line while the green one overlaps with the purple one. In (d), all the lines tend to overlap with each other except the red line. For example, TBNB_FR means that a TBNB classifier with FR feature selection approach.

We contrast the feature selection approaches for the NB and TBNB classifiers. Figure 6.4 (a) shows that the average number of activated features (F_{test}) is lower for the NB classifier across all feature selection approaches than for TBNB with the same feature selection. This shows that the TBNB classifier activates more features for a Twitter user, thereby improving its confidence in the voting intention classification. Unlike in previous work where the OR feature selection approach performs best (Mladenic and Grobelnik, 1999), we find that the WOR and FR feature selection approaches are the most effective in our dataset.

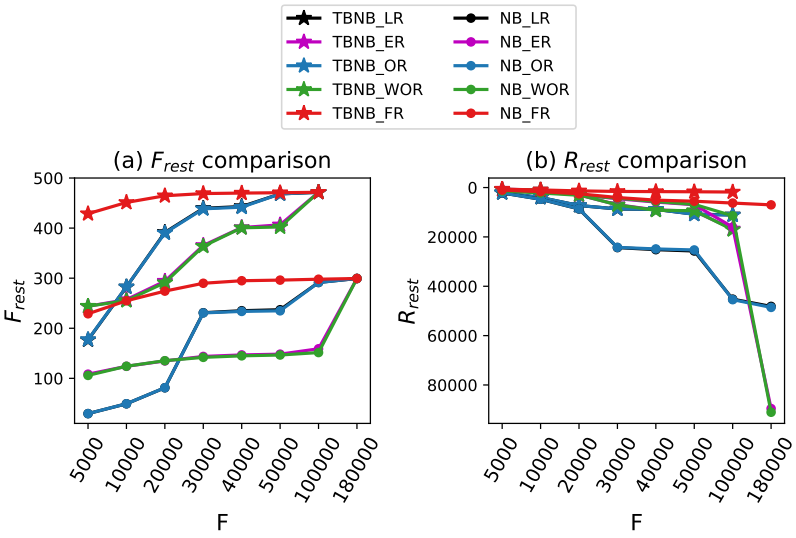

Figure 6.4: Comparison of F_{test} and R_{test} for both the NB and TNBN classifiers ($K = 10$) in our IndyRef community classification task. In both (a) and (b), the blue lines overlap with the black line while the green one overlaps with the purple one.

Next, we consider the features selected and activated by the TBNB and NB classifiers. Firstly, for NB, Figure 6.3 (e) shows that increasing the number of features (F) increases the accuracy, until F reaches an optimal value, and decreases thereafter. The same conclusion is true for TBNB, e.g. for 10 topics (Figure 6.3 (b)). Indeed, we observe from Figure 6.4 (a) that the number of features activated in the unseen tweets (F_{test}) for a given F value is higher for TBNB than for NB, i.e. the classifier has more feature evidence to work with. Moreover, the average rank of those features selected (R_{test}, Figure 6.4 (b)) increases as F increases. Hence, the relatively higher and stable F_{test} and R_{test} values (due to the use of topics) observed for TBNB, in comparison to NB, are indicative of its higher accuracy. Hence, in answering the first research question (i.e. **RQ1** in Section 6.4.3), we conclude that the topic features used in TBNB do indeed improve the accuracy performance of the NB classifier.

In Table 6.10, we report the Precision, Recall, F1 and Accuracy scores of TBNB (K=10 and we use the $20k$ most frequent features with FR feature selection approach[30]) compared to the other commonly used classifiers: RND, DT, SVM, NB and MLP. As can be seen, the performance of the random classifier (RND) is rather limited. We focus on reporting the other classifiers. We use McNemar's test (Fagerland et al., 2013) to compare the performance between two classifiers. In Table 6.10, we use symbols ★, ■, ♣, ♠ and ▽ to index the 5 classifiers. We add an index symbols as a subscript on the "Accuracy" value, if a classifier significantly (p <0.05) outperforms another classifier represented by its index symbol. For

[30]We use these settings as they perform well, as shown in Section 6.4.4.

Table 6.10: The comparisons of 5 classifiers in terms of Precision, Recall and F1 in the IndyRef community classification task. The 5 classifiers are indexed by symbols ★, ■, ♣, ♠ and ▽. If a classifier significantly (p <0.05 in McNemar's test) outperforms another, we add the index symbols as subscripts on the "Accuracy" value of this classifier. The bold values indicate the best performance per column.

Classifiers	Precision		Recall		F1		Accuracy
	Yes	No	Yes	No	Yes	No	
RND	0.727	0.278	0.716	0.288	0.722	0.283	0.599
★DT	0.872	0.658	0.870	0.661	0.871	0.660	0.813
■SVM	0.899	0.669	0.861	0.745	0.879	0.705	$0.829^{\bigstar}$
♣NB	0.925	0.730	0.887	0.810	0.905	0.768	$0.865^{\bigstar,\blacksquare}$
♠MLP	**0.953**	0.720	0.870	**0.886**	0.909	0.794	$0.874^{\bigstar,\blacksquare}$
▽TBNB	0.951	**0.763**	**0.897**	0.878	**0.923**	**0.816**	$\mathbf{0.892}^{\bigstar,\blacksquare,\clubsuit}$

example, the SVM classifier significantly outperforms the DT classifier. First, the TBNB, NB and MLP classifiers are significantly better than both DT and SVM. It seems that the TBNB, NB and MLP classifiers can better deal with the imbalanced dataset[31] in our experiments. The precision and recall of both communities are improved by our TBNB and the MLP classifiers (see the values in columns "Precision" and "Recall" in Table 6.10). Although, there is no significant difference between our TBNB and the MLP classifiers, we can still observe that our TBNB classifier has the highest accuracy score. To summarise, in answering our second research question (**RQ2** in Section 6.4.3), our TBNB approach outperforms the other commonly used classifiers, such as NB (significantly), DT(significantly), SVM (significantly) and MLP (not significantly), on the IndyRef community classification task.

6.4.5 Analysis of the DBpedia Community Classification Task

We list the results of our DBpedia community classification experiments in Tables 6.11 (a)-(c). The classifiers trained using the baseline training dataset (BD), refined baseline training dataset (RBD) and DBpedia training dataset (DBD) are indicated using the subscripts BD, RBD and DBD, respectively. For example, SVM_{RBD} indicates the SVM classifier trained using the refined baseline training dataset. We aim to answer the third research question (namely **RQ3** in Section 6.4.3), i.e. whether the DBpedia labelling approach can generate reasonable datasets for Twitter user community classification, compared to the two baseline labelling approaches (see Section 6.2.2.3). At the same time, we also apply our TBNB approach on the DBpedia training dataset and report the results in Tables 6.12.

[31]The size of "Yes" community is 6326 while it is 2011 for "No" community.

Table 6.11: The community classification results using the three training datasets in the DBpedia community classification task. The rows with the grey background in these three tables indicate the best-performing classifier for a given training dataset. The bold values in (b) indicate the improved performance compared to the best performance in (a) while the bold values in (c) indicate the improved performance compared to the best performance in (b). The superscripts *, † or ⋆ indicate whether the best-performing classifier in (a), (b) or (c) is significantly (p <0.05) outperformed by the others with the superscript.

(a) Using the **baseline training dataset**.

		ACA	MDA	BE	PLT	Accuracy
RDN	F1	0.181	0.344	0.301	0.165	0.261
	Precision	0.159	0.399	0.370	0.136	
	Recall	0.238	0.300	0.253	0.231	
DT_{BD}	F1	0.492	0.0.517	0.291	0.444	0.436
	Precision	0.480	0.527	0.424	0.389	
	Recall	0.505	0.507	0.222	0.517	
NB_{BD}	F1	0.519	0.443	0.614	0.499	0.521
	Precision	0.424	0.507	0.603	0.538	
	Recall	0.641	0.400	0.615	0.460	
SVM_{BD}	F1	0.479	0.435	0.604	0.452	0.510
	Precision	0.383	0.527	0.579	0.583	
	Recall	0.637	0.371	0.632	0.368	
MLP_{BD}	F1	0.488	0.410	0.589	0.429	0.495
	Precision	0.395	0.479	0.576	0.512	
	Recall	0.637	0.358	0.601	0.368	

(b) Using the **refined baseline training dataset**.

RDN	F1	0.203	0.294	0.361	0.165	0.278
	Precision	0.199	0.349	0.360	0.136	
	Recall	0.238	0.250	0.363	0.191	
DT_{RBD}	F1	0.456	**0.528***	0.355	0.376	0.429
	Precision	0.417	0.652	0.333	0.346	
	Recall	0.505	0.444	0.380	0.411	
NB_{RBD}	F1	0.480	**0.531***	0.548	0.417	0.512
	Precision	0.442	0.467	0.740	0.414	
	Recall	0.525	0.616	0.436	0.421	
SVM_{RBD}	F1	**0.547**	**0.515***	**0.653***	0.387	**0.560***
	Precision	0.586	0.503	0.613	0.500	
	Recall	0.512	0.528	0.699	0.316	
MLP_{RBD}	F1	0.514	**0.525***	**0.636**	0.440	**0.553**
	Precision	0.559	0.489	0.634	0.512	
	Recall	0.475	0.566	0.638	0.386	

(c) Using the **DBpedia training dataset**.

RDN	F1	0.198	0.325	0.282	0.173	0.272
	Precision	0.161	0.349	0.361	0.136	
	Recall	0.259	0.304	0.232	0.238	
DT_{DBD}	F1	0.514	0.449	0.613	**0.585***	0.535
	Precision	0.635	0.427	0.536	0.730	
	Recall	0.432	0.474	0.716	0.489	
NB_{DBD}	F1	**0.638**†,*	**0.545***	**0.689**†,*	**0.699**†,*	**0.635**†,*
	Precision	0.500	0.650	0.792	0.570	
	Recall	0.882	0.469	0.609	0.905	
SVM_{DBD}	F1	0.670†,*	**0.566**†,*	**0.683**†,*	**0.727**$^{\dagger,*,+}$	**0.650**†,*
	Precision	0.531	0.645	0.813	0.588	
	Recall	0.906	0.505	0.589	0.952	
MLP_{DBD}	F1	**0.658**†,*	**0.532***	**0.667***	**0.714**†,*	**0.630**†,*
	Precision	0.521	0.615	0.793	0.571	
	Recall	0.894	0.469	0.575	0.952	

The accuracy scores of the DT, NB, SVM and MLP classifiers are markedly better than that of the random classifier RDN (around 0.27). Compared to NB, SVM and MLP, the performance of DT trained using the three training datasets are rather limited in our experiments. Therefore, we focus on evaluating the quality of the three training datasets using three classifiers, NB, SVM & MLP, in Tables 6.11 (a)-(c). The rows with the grey background in these three tables indicate the best-performing classifier for a given training dataset, where the accuracy is used to select the best-performing classifier. Accordingly, for each training dataset, we have one best-performing classifier to represent the quality of the training dataset. First, to check whether the refined baseline training dataset is better than the baseline training dataset, we compare the performance of NB_{RBD}, SVM_{RBD}, MLP_{RBD} to NB_{BD} (the best-performing classifier trained using the baseline training dataset). The superscript "*" in row "F1" indicates that a classifier significantly outperforms NB_{BD}, where the McNemar's test ($p < 0.05$) is used. We find that there are improvements when the Twitter users in the noisy category (c.f. Section 6.2.2.3) are removed. For example, the classifier SVM_{RBD} can perform significantly better in terms of identifying the users in `ACA`, `MDA` and `BE`. This suggests that the refined baseline approach generates a more reliable training dataset than the baseline approach. However, the improvements obtained by the refined baseline approach are still limited. Second, we use the same approach to compare NB_{DBD}, SVM_{DBD} and MLP_{DBD} to NB_{BD} & SVM_{RBD} (the best-performing classifiers trained using the baseline and the refined baseline training datasets). Similarly, we use the superscript "†" in row "F1" in Table 6.11 (c) to indicate that a classifier trained by the DBpedia training dataset significantly performs better than SVM_{RBD}. We find that most of the classifiers trained by the DBpedia training dataset perform significantly better than both the baseline and the refined baseline training datasets, which suggests that our DBpedia labelling approach is more effective than both baseline approaches when generating the ground-truth data for the community classification task. In particular, SVM_{DBD} significantly improves the accuracy score by 9% compared to the best-performing classifier (SVM_{RBD}) trained using the refined baseline training dataset. Indeed, our DBpedia labelling approach can better label `PLT`/`ACA` users which allows the classifiers to have a higher recall performance, e.g. the recall of SVM_{DBD} is 0.95 for `PLT` in Table 6.11 (c) while it is only around 0.3 using the two baseline approaches.

The main reason why the baseline labelling approach cannot generate a good training dataset is that the baseline approach involves too many "noisy" Twitter users. Even when the Twitter users in the noisy category are removed, the performance is still limited due to the small size of the training data. However, our DBpedia labelling approach can generate a larger ground-truth data, which can, to some extent, mitigate the negative effects of these noises when training a classifier. Moreover, it is worth mentioning that our DBpedia

Table 6.12: (a). The classification result of TBNB in the DBpedia community classification task. The superscript *, † or ⋆ indicates that whether TBNB significantly (p <0.05) outperforms NB_{BD}, SVM_{RBD} and SVM_{DBD} (listed in (b)), trained using the three training datasets, respectively.

(a). Applying TBNB on the **DBpedia training dataset**.

		ACA	MDA	BE	PLT	Accuracy
$TBNB_{DBD}$	F1	**0.663** †,*	**0.596**†,*,*	**0.696**†,*	**0.706**†,*	**0.647**†,*
	Precision	0.592	0.624	0.730	0.567	
	Recall	0.752	0.572	0.666	0.936	

(b). The performance of NB_{BD}, SVM_{RBD} and SVM_{DBD}

NB_{BD}	F1	0.519	0.443	0.614	0.499	0.521
	Precision	0.424	0.507	0.603	0.538	
	Recall	0.641	0.400	0.615	0.460	
SVM_{RBD}	F1	**0.547**	**0.515***	**0.653***	0.387	**0.560***
	Precision	0.586	0.503	0.613	0.500	
	Recall	0.512	0.528	0.699	0.316	
SVM_{DBD}	F1	0.670†,*	**0.566**†,*	**0.683**†,*	**0.727**$^{\dagger,*,+}$	**0.650**†,*
	Precision	0.531	0.645	0.813	0.588	
	Recall	0.906	0.505	0.589	0.952	

labelling approach is an automatic labelling approach, which means that the size of the data can be increased easily by using more background data. On the other hand, the refined baseline approach requires human annotators and can be time-consuming. Generally speaking, the accuracy scores from 0.5 to 0.65 are good results for the four communities classification, considering that we only the words in tweets as features in our experiments. More features can be used to improve the performance of our classifiers. For example, Parmelee and Bichard (2011) showed that the reply, mention, and re-tweet features can improve the classification performance for the politician community (discussed in Section 3.4.3). Therefore, there is room for improving these classifiers using different features. In terms of the third research question (i.e. **RQ3** in Section 6.4.3), we conclude that the DBpedia labelling approach outperforms the two baseline labelling approaches. Indeed, it generates ground-truth data that can effectively train a classifier that categorises users into the 4 used communities.

As discussed in Section 6.4.1, we focussed our experiments on the Twitter users in the DBpedia dataset. However, to show the generalisation of our DBpedia labelling approach, we report the performance of the trained classifiers in Table A.4 in the Appendix, where the Twitter users from the refined baseline dataset are used as the test data. The results obtained on the refined dataset show that our DBpedia labelling approach still generates effective ground-truth data for training an accurate user community classifier. Next, we report the performance of our TBNB approach for the DBpedia community classification task.

We apply our TBNB approach on the DBpedia training dataset[32]. Table 6.12 (a) lists the result of our TBNB classifier trained using the DBpedia training dataset. Our TBNB

[32]We find that the topic number K in our TBNB approach does not affect the classification performance much in the DBpedia community task. Hence, we set 20 topics in our TBNB approach for the DBpedia community classification.

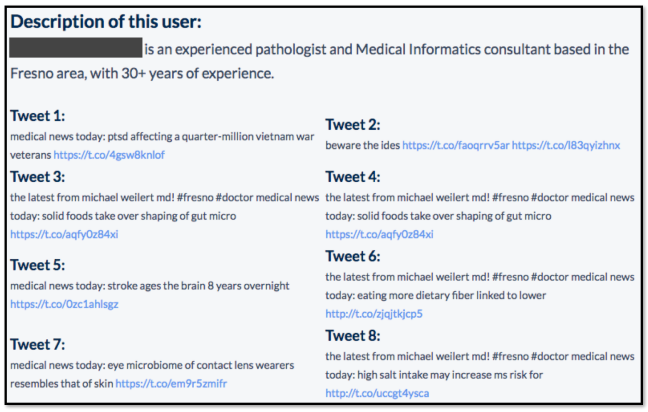

Figure 6.5: Example of a Twitter user.

classifier significantly outperforms the classifiers trained using the baseline and refined baseline training datasets (see the accuracy $0.647^{\dagger,*}$ in Table 6.12 (a)). On the other hand, the TBNB approach does not outperform the SVM classifier, i.e. SVM_{DBD} (see Table 6.12 (b)). However, their performance are very close (i.e. neither of them significantly outperforms the other).

In terms of the MDA community, TBNB_{DBD} outperforms SVM_{DBD}. This indicates the high effectiveness of our TBNB approach. It is worth mentioning that the MDA community is the most difficult community for the classifiers to identify (the F1 scores in column "MDA" are the lowest compared to others in Table 6.12 (b)). This is because users belong to other communities can be misclassified to MDA. To illustrate, for a Twitter user shown in Figure 6.5, words "pathologist" and "medical" indicate that this user belongs to ACA. However, a classifier can identify this users as MDA because of words "latest" and "news".

The reason why TBNB does not outperform the best classifier (i.e. SVM_{DBD}) might be that the four communities discussed topics using similar usages of words. As discussed in Section 6.3, the "Yes" and "No" voters discussed the topic "oil" from different perspectives in IndyRef and therefore the usage of words is different in these two communities (denoted by the notion of dissimilarity, introduced in Section 6.3). However, in the DBpedia training dataset, the Twitter users might not hold different opinions about a topic and therefore the dissimilarities of word features can be low. Table 6.13 lists 3 topic examples that are used in TBNB on the DBpedia training dataset. Indeed, these topics are not controversial so as to draw orthogonal discussions among communities. Hence, the improvements brought up by TBNB on the DBpedia training dataset might have been limited.

In summary, although TBNB does not perform the best in our DBpedia community classification task, its performance is very comparable to the best one (i.e. SVM_{DBD}). Moreover, our TBNB approach significantly outperforms the remaining three classifiers (i.e. NB,

Table 6.13: Topics and associated words in the DBpedia training dataset. For each topic, the top 10 ranked words are listed in column "Associated Words".

Topic	Associated Words
1	social, news, media, mp, bbc, world, people, story, break, youtube
2	apple, win, best, market, stock, ebay, amazon, time, world, business
3	follow, new, free, watch, want, like, love, year, today, share

DT and MLP) in the DBpedia community classification task, thereby showing a parallel with its performance in the IndyRef community classification task (c.f. Section 6.4.4). This answers our fourth research question (**RQ4** in Section 6.4.3), i.e. the results of TBNB on the IndyRef dataset do seem to generalise to the DBpedia dataset.

6.5 Conclusions

In this chapter, we have investigated how to identify communities (i.e. 'who') during a political event on Twitter. We first proposed two ground-truth generation approaches: the hashtag labelling and the DBpedia labelling approaches. We showed how the hashtag labelling approach can generate ground-truth data for classifying Twitter users into communities during an election or a referendum while the DBpedia labelling approach can generate ground-truth data for classifying communities according to the Twitter users' professions. To evaluate the hashtag labelling approach, we used the Twitter users' followee network to check how the labelled Twitter users follow the members of various parties. The results showed that our hashtag labelling approach had a high agreement with the followee network verification method, which suggests that our hashtag labelling approach was effective (c.f. Table 6.1). To evaluate the DBpedia labelling approach, we conducted a crowdsourced user study and showed that our DBpedia labelling approach has a good-level agreement with human judgements (c.f. Table 6.6).

We also proposed a Topic-based Naive Bayes (TBNB) to classify the community affiliations of Twitter users. Our proposed TBNB approach leveraged the dissimilarity of the features across the discussed topics, and their variance across the communities to improve the performance of the Twitter user community classifier. We conducted our classification experiments on both the IndyRef community and the DBpedia community datasets in order to evaluate our TBNB approach. We showed that our TBNB approach outperformed Naive Bayes (significantly), Decision Trees (significantly), Support Vector Machines (significantly) and Multilayer Perception (not significantly) (see Table 6.10) in the IndyRef community classification task. On the other hand, our TBNB approach performed second-best (see Tables 6.11

and 6.12) in the DBpedia community classification task. Overall, these results taken together, suggest that our TBNB approach is promising and effective for classifying Twitter users into communities. We showed that our two ground-truth generation approaches can be used to train classifiers with reasonable performance, which indicates that they can be applied to effectively generate ground-truth data for the Twitter user community classification task.

So far, we have investigated topic coherence metrics, a time-sensitive topic modelling approach and approaches for classifying Twitter users into communities. In order to demonstrate the generalisation of our proposed approaches, we will apply these proposed approaches to identify communities and extract coherent topics during the US Election 2016 event in the next chapter.

Chapter 7

Application on US Election 2016

7.1 Introduction

In Chapters 4, 5 and 6, we have investigated various topic coherence metrics, a time-sensitive topic modelling approach and approaches for classifying Twitter users into communities. In this chapter, we aim to investigate the generalisation of the results obtained from the previous chapters on a large different political event-related dataset, i.e. a Twitter dataset pertaining to the US Election 2016 event. We not only assess the performance of our approaches in the US Election-related dataset, but also check with a social scientist whether our approaches are useful when analysing this key political event.

To analyse US Election 2016 on Twitter, we make use of our approaches to first identify Twitter users into two communities (i.e. identifying the 'who') that support the presidential candidate Donald Trump (i.e. the "proTrump" community) and Hillary Clinton (i.e. the "pro-Clinton" community), respectively, and then extract their discussed topics (i.e. 'what'). We focus on these two communities because they are key to any political analysis of this event. Figure 7.1 shows how we apply our proposed approaches in an application on US Election 2016. Specifically, first, we apply our proposed hashtag labelling approach (see Chapter 6) to generate ground-truth data (i.e. the labelled dataset), which allows to train a user community classifier. Second, we apply our Topic-based Naive Bayes (TBNB, see Chapter 6) approach to identify which community of the two that a given Twitter user belongs to. To evaluate our TBNB classifier, we also conduct a user study to obtain human ground-truth labels of the Twitter users sampled from the unlabelled data. The trained classifier allows to categorise our election-related data into the two communities. Third, to examine what topics the pro-Trump and proClinton communities discussed, we apply our time-sensitive topic modelling (TVB, see Section 5) approach to extract the topics from the two identified communities tak-

ing into account the time dimension of tweets (i.e. 'when'). To evaluate the generated topic models and select the most coherent topics, we apply our proposed Twitter topic coherence metrics (see Chapter 4). Moreover, we compare the topics generated by using our TVB approach with those generated by using the classical LDA approach in terms of interpretability. With the help of a social scientist, we investigate whether TVB generates more interpretable topics than those generated by using the classical LDA approach. Finally, we examine the extracted community-related topics to analyse the behaviours of the two communities during the election. The remainder of this chapter is organised as follows:

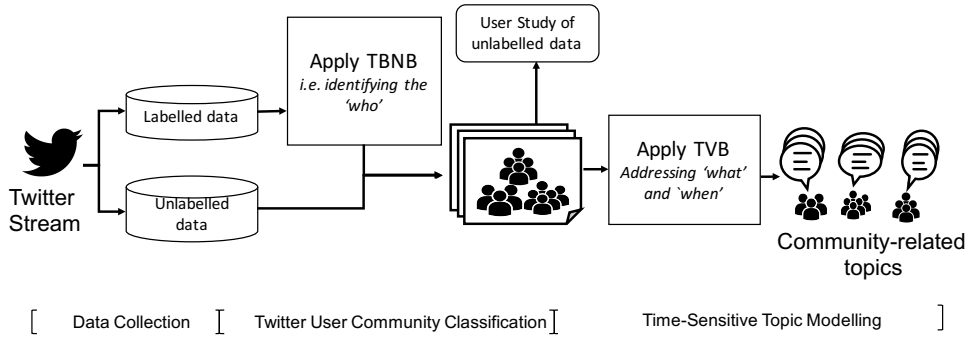

Figure 7.1: Our application on US Election 2016.

- Section 7.2 describes how we collect our data including labelled and unlabelled datasets.
- Section 7.3 aims to evaluate our TBNB approach on US Election 2016. We also assess the effectiveness of the hashtag labelling approach for training a user community classifier. This section verifies the generalisation of the results obtained in Chapter 6.
- Section 7.4 prepares the data for extracting community-related topics, i.e. we apply the trained TBNB classifier to categorise Twitter users into the two communities that social scientists are interested in: proClinton and proTrump.
- Section 7.5 describes how we apply our TVB approach to extract topics discussed from the two identified communities. We aim to evaluate our TVB approach using our proposed Twitter topic coherence metrics. This section verifies the generalisation of the results obtained in Chapter 4.
- In Section 7.6, we confirm whether our proposed approaches assist social scientists. We analyse the interpretability and usefulness of topics generated by TVB. We also compare our generated topics using TVB to those obtained using the classical LDA. This section verifies the generalisation of the results obtained in Chapter 5.
- Section 7.7 provides the conclusions of this chapter.

7.2 Data Collection

We collect a sample of tweets posted in the US within a three month period leading up to the election, from 01/08/2016 to 08/11/2016 (election day). This Twitter dataset is crawled using the Twitter Streaming API[1] by setting a bounding box to cover only the area of the US, which can obtain a sample of roughly 1% of all tweets in the US (according to Morstatter et al., 2013)[2]. We use the bounding box because it allows us to obtain tweets that are posted within the US[3] and since we are interested in the views of the US Twitter users rather than users from other parts of the world. The collected tweets either have exact geo-locations (i.e. longitude & latitude) or have place information (e.g. New York City) identifiable by Twitter. We collect approximately 1.5 million tweets per day. In total, we obtain a sample of roughly 150 million tweets. In the following sections, we generate our labelled and unlabelled datasets from the collected 150 million tweets (discussed further in Sections 7.2.1 and 7.2.2, receptively). The labelled dataset allows us to train and test[4] a user community classifier, which can then be applied to categorise the Twitter users in the unlabelled dataset into two communities: i.e. the proTrump and proClinton communities. Next, we describe how to obtain the labelled and unlabelled datasets.

7.2.1 Labelled Data — Applying the Hashtag Labelling Approach

To obtain the labelled dataset (i.e. ground-truth data) for training the user community classifier for US Election 2016, we apply our hashtag labelling approach. We first identify a set of hashtags signalling vote preference during the election, which are listed in Table 7.1. As can be seen from Table 7.1, our chosen hashtags signal support in clear ways. Moreover, these hashtags were widely used by users during the election, which allows us to obtain a large labelled dataset, i.e. our ground-truth data.

Similar to our methodology for the Scottish Independence Referendum 2014 (c.f. Section 6.2.1.1), we identify Twitter users who consistently use the hashtags of proTrump (i.e. only use the hashtags listed in column"proTrump" of Table 7.1) and those who consistently use the hashtags of proClinton ("proClinton" in Table 7.1) from the collected 150 million tweets. We then obtain a labelled dataset containing 39,854 users who produced 394,072 tweets, as shown in Table 7.2. Note that retweets are not included to avoid repeating tweets in

[1] `https://dev.twitter.com`

[2] Our dataset includes tweets posted from Alaska but not Hawaii.

[3] We rely on Twitter's internal process to determine whether the tweet has been posted in the US or not. More information is provided at `https://developer.twitter.com/en/docs/tutorials/filtering-tweets-by-location.html`.

[4] We apply 10-fold cross-validation to evaluate a classifier on the labelled dataset.

Table 7.1: The used hashtags for labelling Twitter users in US Election 2016.

proClinton	proTrump
#imwithher	#trumptrain
#alwayshillary	#alwaystrump
#strongertogether	#votetrump
#nevertrump	#crookedhillary
#dumptrump	#neverhillary
#notrump	#corrupthillary
#antitrump	#nohillary

Table 7.2: The labelled and unlabelled datasets for the user community classification of US Election 2016.

Datasets	Number of users			Number of tweets		
	proClinton	proTrump	Total	proClinton	proTrump	Total
Labelled Dataset	28,168	11,686	**39,854**	245,692	148,380	**394,072**
Unlabelled Dataset	unknown	unknown	**224,664**	unknown	unknown	**3,171,264**

our ground-truth data. Specifically, our hashtag labelling approach generates 28.1k users in the proClinton community who posted 245.6k tweets, and 11.6k users in the proTrump community who tweeted 148.3k times, as seen in Table 7.2. Note that, in our labelled dataset, the number of users in the proClinton community is larger than that in the proTrump community.

7.2.2 Unlabelled Data

For our unlabelled dataset, we sample tweets from the collected 150 million tweets, which contained either keywords or hashtags (or both) that we consider election-related. For example, we have tweets with keywords or hashtags such as "Trump"[5] or"Hillary" or "Clinton" or "debate" or "vote" or "election". The used election-related hashtags are #clinton, #trump, #hillary, and #debatenight[6]. We then choose all the tweets from all users who posted at least 4[7] tweets that used such election-related keywords or hashtags from the collected 150 million tweets. In total, we have 224,664 users with 3,171,264 tweets in our unlabelled dataset, as shown in Table 7.2. Note that our unlabelled dataset does not contain any Twitter users in our labelled dataset.

To summarise our data collection processes, we show how we generate our labelled and unlabelled datasets for our application on US Election 2016 in Figure 7.2. We also explain how we use the collected labelled and unlabelled datasets in Table 7.3. We first collect a sample of roughly 150m tweets posted in the US. We apply the hashtag labelling approach

[5]The keyword "Donald" is not selected as we found that it introduces too much noise, as it is more generic than "Trump" or "Hillary" for example. We did not want to collect tweets about "Donald Duck" for instance.

[6]Note that the case sensitivity of these hashtags is omitted, i.e. #TRUMP is the same as #trump.

[7]Including users who tweet only one or a few times can introduce too much noise into the analysis.

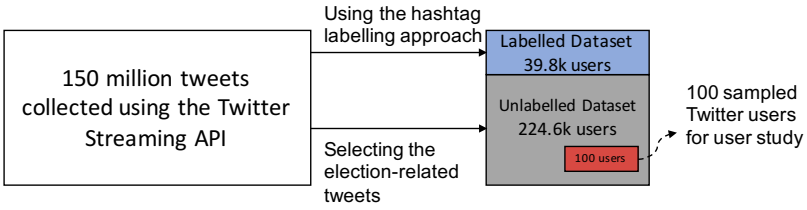

Figure 7.2: Our labelled and unlabelled datasets for US Election 2016.

to generate a labelled dataset containing about 39.8k users (see row "Labelled Dataset" in Table 7.2 and the blue box in Figure 7.2). We use this labelled dataset to train our TBNB classifier and evaluate it in comparison to the commonly used classifiers. Second, we select the tweets using the election-related keywords and obtain an unlabelled dataset containing approximately 224.5k users (see row "Unlabelled Dataset" in Table 7.2 and the grey box in Figure 7.2). Our unlabelled dataset is much larger than our labelled dataset, given that our labelled dataset includes only users who used hashtags (in Table 7.1) consistently and their respective tweets. The community affiliations of the Twitter users in the unlabelled dataset are what we aim to determine. As mentioned in Section 7.1, we sample 100 Twitter users (the red box in Figure 7.2) from our unlabelled dataset. We then conduct a user study to obtain the human ground-truth labels for these 100 users, which are used to identify whether the community affiliations determined by a given classifier align with human judgements. Moreover, we apply our trained TBNB classifier (from Chapter 6) on the unlabelled dataset (without the 100 sampled Twitter users) and categorise the Twitter users into two communities: i.e. proClinton and proTrump. To examine the discussed topics from the two identified communities, we apply our TVB approach (from Chapter 5) to extract topics from the tweets of the two communities in both the labelled and unlabelled datasets.

Table 7.3: The use of the labelled and unlabelled datasets in our application.

Datasets	Used for	Described in
Labelled dataset	Evaluating TBNB approach	Section 7.3
100 sampled users from the unlabelled dataset	User study	Section 7.3
Unlabelled dataset without the 100 sampled users	Categorising users into the two communities using TBNB	Section 7.4
Labelled and unlabelled datasets	Extracting topics from the two communities using TVB	Section 7.5

It is worth mentioning that recent attention has been drawn to the role of fake news and Twitter bot accounts in influencing public opinion, particularly fake news and bots originating from Russia during US Election 2016 (Allcott and Gentzkow, 2017; Guess et al., 2018; Soroush et al., 2018; Howard et al., 2018; Timberg, 2016). To ascertain the presence of Russian bots in our analysis, we turn to a list of 2752 Russian bot accounts that were identified by the US House Select Committee on Intelligence[8]. We then examine how many

[8]These Twitter accounts can be downloaded from `https://democrats-intelligence.house.`

tweets from these accounts exist in our labelled and unlabelled datasets. We found that none of these Russian bot accounts is present in our labelled dataset, and a mere 25 tweets from 16 Russian bots are present in our unlabelled dataset. Thus, we argue that the influence of these identified bot accounts on our analysis is minimal. Our use of a bounding box for our data collection that restricted tweets to accounts within the US might be the reason why we find so few tweets from these Russian bot accounts in our data. In the next section, we first evaluate our TBNB approach on US Election 2016.

7.3 Evaluating TBNB on US Election 2016

In this section, we aim to evaluate our proposed TBNB approach on US Election 2016 in comparison to the commonly used classification approaches. We also investigate whether our proposed hashtag labelling approach can effectively generate ground-truth data for training a user community classifier. To evaluate the performance of our TBNB approach, we conduct our classification experiments using the labelled dataset where cross-validation is applied. As mentioned in Section 7.2, we also conduct a user study to obtain human ground-truth labels for the 100 sampled users, which are used to identify the agreement between human assessors and a classifier when categorising the users in the unlabelled dataset into the two communities. We aim to answer the following research questions:

- **RQ1.** Can our proposed TBNB classifier perform better than the other commonly used classifiers on the labelled dataset?
- **RQ2.** Can our proposed hashtag labelling approach effectively generate a ground-truth data for training a user community classifier?
- **RQ3.** Can our proposed TBNB classifier align with the human assessors when classifying the users in the unlabelled dataset?

Next, we first describe our classification experimental setup in Section 7.3.1 and the conducted user study in Section 7.3.2. We then report the results of our user community classification experiments and our user study in Sections 7.3.3 and 7.3.4, respectively.

7.3.1 Experimental Setup

We apply our proposed TBNB classification approach in our community classification task for US election 2016. There are two classes: proClinton and proTrump. Similarly to Chapter 6, we also apply the commonly used classifiers for comparison: the random classifier

gov/uploadedfiles/exhibit_b.pdf

(RDN), Decision Trees (DT), Naive Bayes (NB), Support Vector Machines (SVM) and Multilayer Perceptron (MLP). We set the number of topics K in TBNB to 10 and use 100 neurons in MLP[9]. The details setup of these commonly used classifiers are the same as their setup in Section 6.4.2.1. We use the 5000[10] most frequent words as features to train our classifiers. Twitter users are translated into TF−IDF vectors as input to the classifiers. We apply a 10-fold cross-validation for all classifiers using the labelled datasets (39.8k users in Table 7.2) to evaluate the performance of these classifiers, i.e. our labelling dataset is partitioned to 10 subsamples. In each fold, one subsample is used as test data while the rest of the 9 subsamples are used as training data. Similarly to Section 6.2.1.1, we remove all the hashtags (see Table 7.1) using in our hashtag labelling approach from our labelled dataset. Since we find that the proTrump community (11.6k users) is smaller than the proClinton community (28.2k users) (see Table 7.2) in our labelled dataset, we apply oversampling to the proTrump community to avoid class imbalance that may bias the learned classifiers. To evaluate the performance of our classifiers for each community, we use four metrics: Precision, Recall, F1 and Accuracy (also used in Chapter 6).

7.3.2 User Study for Twitter users' Candidate Preferences

We aim to classify the community affiliations of the Twitter users in the unlabelled dataset. Since there are no community labels in the unlabelled dataset, we cannot evaluate the performance of our classifiers (trained using the labelled dataset) on the unlabelled dataset. Therefore, as mentioned above, we sample 100 Twitter users from our unlabelled dataset and conduct a user study to obtain the human ground-truth labels for the 100 Twitter users, which allows us to evaluate the performance of a trained classifier on the unlabelled dataset. We use Crowdflower platform (also used in Chapters 4 and 6) to conduct our user study. We ask Crowdflower workers to determine whether a given Twitter user supports Hillary Clinton or Donald Trump (or neither) by looking at the content of the user's tweets, for the 100 sampled Twitter users[11]. Thus we identify the agreement between our trained classifiers (our TBNB classifier and the other commonly used classifiers mentioned in Section 7.3.1) and human assessors in identifying the community labels of the 100 Twitter users.

[9] In our preliminary experiments, we set different numbers of topics (K) in TBNB and we found that the TBNB classifier performed well with K=10. For the same reason, we set the number of neurons to 100 in MLP.

[10] We showed that the 5000 most frequent words were effective to train classifiers in the Scottish Independence Referendum 2014 (see the red lines in Figure 6.3). We do not use a higher number of features because we have a large size of the unlabelled dataset and a higher number of features can increase the computational cost.

[11] Note that we did not disclose the user's account handle, or name, nor any other identifying information.

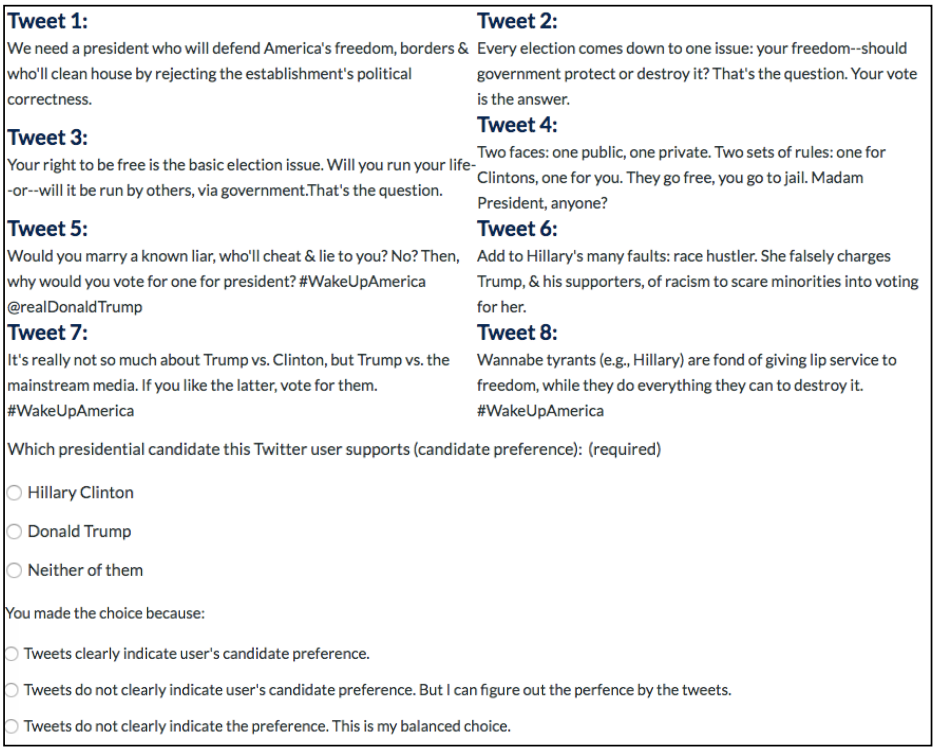

Figure 7.3: The user interface of our Crowdflower user study for US Election 2016.

For each of the 100 selected Twitter users, we present crowdsourced workers with at most 8[12] of their respective tweets selected randomly, as seen in the top half of Figure 7.3. After reading the 8 tweets, a Crowdflower worker is asked to select whether the given Twitter user supports Hillary Clinton, Donald Trump, or neither of them, as seen in the middle of Figure 7.3. To understand how the workers reach their decisions, we also ask them to explain their reasoning through three provided choices: 1) "Tweets clearly indicate the user's candidate preference"; 2) "Tweets do not clearly indicate the user's candidate preference". However, they can figure out their preferences from the tweets; 3) "Tweets do not clearly indicate the preference". This is their balanced choice (see the bottom of Figure 7.3). The Crowdflower workers are required to spend at least 20 seconds[13] for each judgement. Each worker is paid $0.20 for each judgement. Similarly to the user study in Section 6.2.2.5, to ensure quality, we prepare a set of test questions, where the community labels of the Twitter users are verified in advance. CrowdFlower workers can only enter the task if they reach 70% accuracy on the test questions. We obtain 3 independent judgements of whether each of our 100 Twitter users was proClinton or proTrump, or neither[14]. We report the results of this user study in Section 7.3.4.

[12]We used at most 8 tweets to make the task more manageable and feasible.

[13]Crowdflower allows to set the minimum time for each question to control the quality of the user study.

[14]Note that our classifier identifies users into either a proClinton or a proTrump community and does not include a third option of "neither".

7.3.3 Results of User Classification Experiments

Table 7.4 lists the results[15] of our classification experiments using the labelled dataset. Similarly to the experiments in Section 6.4.4, we use McNemar's test to see whether two classifiers perform equally. From Table 7.4 we can see that, with the exception of the random classifier (RDN), all of the classifiers exhibit a strong performance on the F1, Precision, Recall and Accuracy metrics. It is clear that Twitter users in the proClinton and proTrump communities differentiated themselves well from one another, which suggests that the language of their tweets was sufficiently distinct so as to be able to classify users correctly as proClinton and proTrump in ways consistent with their adoption of the hashtags displayed in Table 7.1. We observe that the TBNB classifier achieves the highest accuracy among all the classifiers. To answer the first research question (i.e. **RQ1** in Section 7.3), we find that our TBNB approach can outperform the other commonly used classifiers, DT (significantly), SVM (significantly), MLP (significantly) and NB (not significantly according to the McNemar's test), on the labelled dataset of US Election 2016. Our TBNB approach is able to classify the community affiliations of 85.1% of the Twitter users in our labelled dataset accurately using the words used in their tweets. After TBNB, the NB classifier performs second best in our labelled dataset. The performance of our TBNB classifier aligns with its performance in Chapter 6, where we have shown that our TBNB classifier performed best on a referendum-related dataset and second best on a 4-community classification dataset, among the 5 trained classifiers (i.e. DT, NB, SVM, MLP and TBNB, see Section 6.4). In the next section, we identify whether our TBNB classifier can identify the community affiliations of the 100 Twitter user sampled from the unlabelled dataset, also compared to the rest of classifiers.

7.3.4 Results of User Study

In our user study, we obtain 336 judgements from 31 different workers for labelling the 100 Twitter users sampled from our unlabelled dataset. Among the 100 users, 76 users are labelled as either proClinton or proTrump according to the online workers. Among these 76 Twitter users, the crowdsourced workers were unanimous for 51 (67%), meaning that all three workers agreed that the Twitter user was proClinton, or all three agreed that the user was proTrump. Concerning their explanations for how they determined whether a Twitter user was proClinton or proTrump, for 31 users, the workers marked that the "Tweets clearly indicate the user's candidate preference"; for 42 Twitter users the workers answered that the "Tweets do not clearly indicate the user's candidate preference. However, I can figure out

[15]These results are obtained by applying 10-fold cross validation for all classifiers.

Table 7.4: The Twitter user community classification results on US Election 2016. These results are obtained using our labelled dataset, where a 10-fold cross-validation is applied for all classifiers. We highlight in bold the highest values for reference.

		Candidate Community		
		proClinton	proTrump	Accuracy
RDN	F1	0.498	0.499	0.499
	Precision	0.499	0.499	
	Recall	0.497	0.500	
DT	F1	0.817	0.639	0.757
	Precision	0.874	0.567	
	Recall	0.768	0.733	
NB	F1	0.883	**0.760**	0.843
	Precision	**0.930**	0.689	
	Recall	0.840	**0.849**	
SVM	F1	0.881	0.747	0.838
	Precision	0.916	0.690	
	Recall	0.848	0.814	
MLP	F1	0.835	0.678	0.782
	Precision	0.897	0.597	
	Recall	0.781	0.784	
TBNB	F1	**0.893**	0.753	**0.851**
	Precision	0.903	**0.734**	
	Recall	**0.883**	0.772	

Table 7.5: The agreement between classifiers and human assessors on US Election 2016.

Classifier	Cohen's $kappa$	Accuracy
RDN	-0.013	0.50
DT	0.44	0.72
NB	0.60	0.80
SVM	0.62	0.82
MLP	0.58	0.79
TBNB	**0.66**	**0.83**

the preference by the tweets"; and for 3 Twitter users, the workers selected that the "Tweets do not clearly indicate the preference. This is my balanced choice."

We train our classifiers using the labelled dataset as training data. We then applied these trained classifiers on the 76 sampled Twitter users[16] and obtain the community affiliations of these 76 Twitter users. The obtained community affiliations by the classifiers are compared to the human ground-truth labels of these 76 users. Table 7.5 displays the Cohen's $kappa$ and accuracy scores of the classifiers compared to the human assessors. All classifiers (with the exception of RDN) achieve reasonable accuracy scores. This finding answers our second research question (i.e. **RQ2** in Section 7.3), i.e. our proposed hashtag labelling

[16]We remove the other 24 Twitter users since human assessors did not have an agreement on them.

approach can generate effective ground-truth data for training a valid and reliable user community classifier. Among these classifiers, our TBNB classifier performs significantly better than the other commonly used classifiers excepting SVM according to McNemar's test. We can see that our TBNB classifier has higher $kappa$ and accuracy scores than the others, consistent with what we saw with the labelled dataset (see Section 7.3.3). This answers our third research question (i.e. **RQ3** in Section 7.3), i.e. our TBNB classifier best aligns with the human assessors when classifying the Twitter users in the unlabelled dataset.

The remaining 24 users of the 100 sampled users do not have clear community affiliations according to the CrowdFlower workers. A possible reason is that we present crowdsourced workers with Twitter users and their 8 recent tweets. However, 8 tweets might not be enough for crowdsourced workers to identify the community affiliations of the given Twitter users. Examining additional content might be helpful. In addition, our TBNB classifier uses the top 5000 words as features, which is far more than any Crowdflower worker sees among 8 tweets.

7.4 Applying TBNB on US Election 2016

In this section, we apply our trained TBNB classifier on the unlabelled dataset[17] so as to categorise the Twitter users into the two proClinton and proTrump communities and prepare the data for extracting the community-related topics in the next section. We use TBNB due to its high effectiveness, as shown in Section 7.3. To categorise the Twitter users in the unlabelled datasets, we train our TBNB classifier using the whole labelled dataset as training data[18]. Next, we first show the number of the classified Twitter users/tweets in the two communities on our unlabelled dataset. Second, we show the number of tweets posted in the proClinton and proTrump communities over time in the three months' time period, which can be seen as indicators of the popularity of these two candidates over time.

Table 7.6: The number of users/tweets of the proClinton and proTrump communities in the unlabelled dataset.

Dataset	Number of users			Number of tweets		
	proClinton	proTrump	Total	proClinton	proTrump	Total
Unlabelled Dataset	168,534	56,130	**224,664**	1,880,584	1,290,680	**3,171,264**

As shown in Table 7.6, we observe that our labelled dataset contains 168,534 proClinton users authoring 1,880,584 tweets (11.2 on average) and 56,130 proTrump users with

[17]The 100 sampled Twitter users are excluded since their community affiliations are already identified by the human assessors.

[18]The setup of TBNB is still the same as described in Section 7.3.1

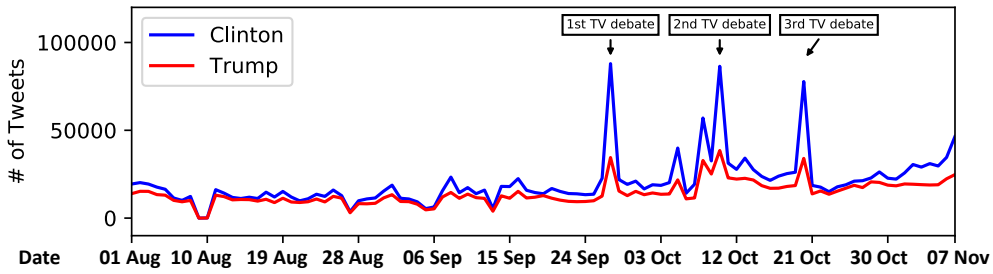

Figure 7.4: The number of tweets from the proClinton and proTrump communities over time in US Election 2016.

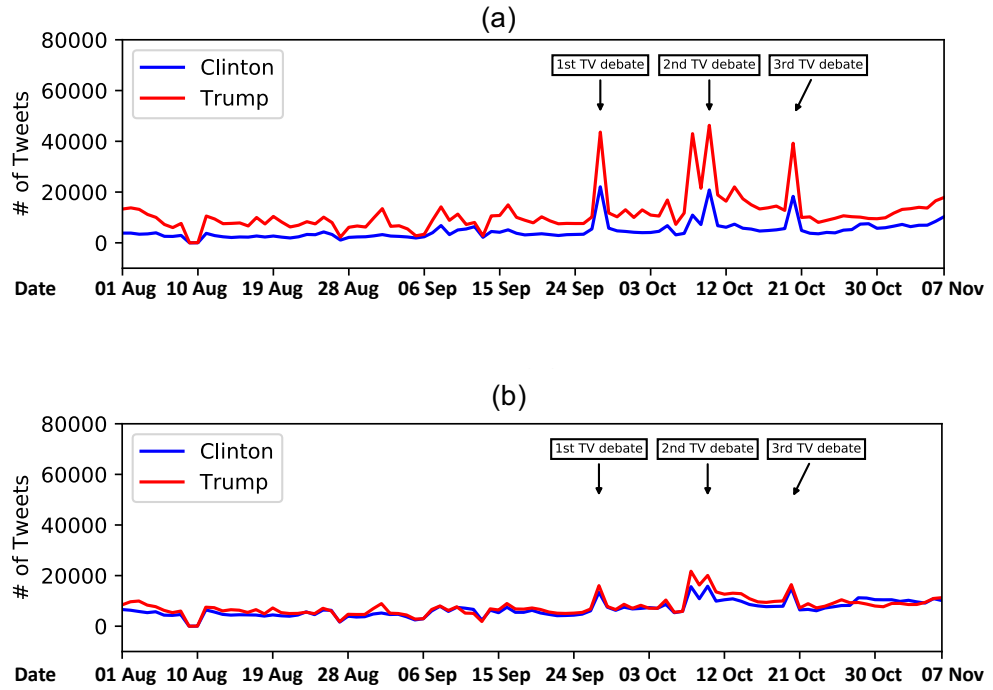

Figure 7.5: The number of tweets mentioning two candidates in the (a) proClinton and (b) proTrump communities in US Election 2016.

1,290,680 tweets (22.0 tweets on average). It means that the proClinton community across the Twitter platform is much more sizable than the proTrump one in terms of the number of users and the number of election-related tweets, but the Trump supporters tweet more often on average.

Since each tweet is associated with a time point, we can also examine the dynamics of support, first overall, and then by community. In Figure 7.4, we show the number of tweets that were posted by the proClinton and proTrump communities over time. Not only were proClinton tweets more plentiful as we showed in Table 7.6, but they were more prolific over the entire period of analysis. During the three US televised debates, marked by spikes in the data, we see a particular activity among the proClinton community.

We also compare the use of the words "Clinton" and "Trump" among the proClinton and proTrump communities, as shown in Figure 7.5. We note that this display is simply for a descriptive comparison as we focus on the "Clinton" and "Trump" word usage, and do not include "Hillary" for example. Figure 7.5 (a) represents tweets in the proClinton community, whereas Figure 7.5 (b) reflects tweets in the proTrump community. From this visualisation, we see that in both the proClinton and proTrump communities, the word "Trump" was more common most of the time and especially so in the proClinton community. This suggests that the proClinton community focused more on invoking their opposing candidates' last name than their own candidate's.

In this section, we have classified the Twitter users in the unlabelled dataset into the two communities (see Table 7.6). In the next section, we apply our TVB approach to extract topics discussed in these two communities.

7.5 Applying TVB on US Election 2016

In this section, we apply our TVB approach to extract topics discussed in the proClinton and proTrump communities. We aim to evaluate the coherence of the generated topic models and choose the most coherent topic models for analysis. In our experiments, we vary the number of topics (K) in order to obtain topic models with a high coherence. As observed in Chapter 4, the coherence of topics increases when K increases. We verify this conclusion in this section. On the other hand, we aim to choose the topic models with the best K. We aim to answer the following two research questions:

- **RQ4.** Does the coherence of the generated topics increase when K increases?
- **RQ5.** How to select the best K?

Next, we first describe our topic modelling experimental setup in Section 7.5.1. We then report the coherence of the generated topic models in Section 7.5.2.

7.5.1 Experimental Setup

For each community, we extract topics discussed by its Twitter users in both labelled and unlabelled datasets. We include the Twitter users in the labelled dataset in our topic modelling experiments since there are also a large number of Twitter users in the labelled dataset and we are interested in what these Twitter users said about the election. We sample $200k$ tweets posted by Twitter users from each community in both the labelled and unlabelled datasets. We then apply our TVB approach on the $200k$ sampled tweets from each community to extract the discussed topics, since TVB has been previously shown to be effective for Twitter

data in Chapter 5. The number of topics, K, has implications on the coherence of the topics that are extracted (Section 4.6). To select K, we set K from 10 to 100, with step 10 in order to obtain topic models with a good quality. We set the balance parameter δ=0.8 in TVB since we show that δ=0.8 works well in Chapter 5. The other setup of TVB is the same as its setup in Section 5.4.2. To evaluate the coherence quality of the resulting topics, we use the best WE-based Twitter coherence metric proposed in Chapter 4, namely T-WE$^{w=3}_{d=500}$ [19]. We use both the average coherence and `coherence at n` (`c@n`) (c.f. Section 4.6.1) to evaluate the coherence of the generated topics and topic models ,where n is set to $\{10, 20, 30\}$[20], i.e. evaluating the top 10/20/30 most coherent topics of the generated topics.

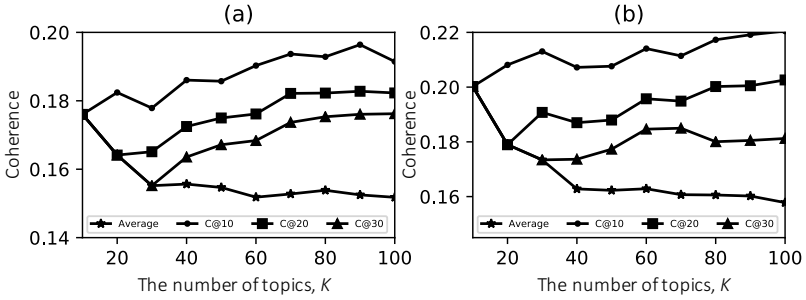

Figure 7.6: The coherence of topic models with different K. (a) topic models generated from the proClinton-related tweets; (b) topic models generated from proTrump-related tweets.

7.5.2 Coherence Results

We show the coherence results of the generated topic models from proClinton and proTrump, in Figure 7.6 (a) and (b), respectively. First, it is clear that the coherence of the generated topics increases when K increases. For example, we can observe that the `c@10` and `c@20` coherence scores increase in both communities when K increases. This answers our fourth research question (i.e. **RQ4** in Section 7.5), i.e. a topic model can have more coherent topics with a larger K. When choosing the best K, we expect that a topic model can have a high coherence. Meanwhile, the best K should not be too big since a bigger K means more topics and they require more time when examining their content. Therefore, we increase the values of K in our experiments and choose the best K when TVB starts to generate topic models with a stable coherence. This can ensure that we can have a topic model with a high coherence and the chosen K is not too big. For proClinton, we choose the best K as 70

[19]Note that the WE-based metric is also used in Chapter 5. The used WE-based metric in this chapter is the same as the one used in Section 5.4.3.1 since the time period of the used Twitter background data covers a long time period before the election date.

[20]The setting of n=$\{10, 20, 30\}$ for `c@n` is reasonable considering that the largest number of topics in our experiments is 100.

since the coherence of topic models in proClinton starts to become stable when K=70 (see the lines of `c@20` and `c@30` in Figure 7.6 (a)). On the other hand, for proTrump, we choose K=60 for the same reason. This answers our fifth research question (i.e. **RQ5** in Section 7.5), i.e. we choose the best K, which is not too big and which also allows TVB to generate topic models with a high and stable coherence. In the next section, we examine the topics in the topic model with K=70 in proClinton and the topic model with K=60 in proTrump in order to analyse US Election 2016 on Twitter.

7.6 Analysing Topics in US Election 2016

In this section, we aim to check with a social scientist whether our topics generated using TVB are interpretable and useful and whether these topics are more interpretable and useful than those generated using LDA. We first present a social scientist[21] with the topics generated from the two identified communities using TVB. We also present the social scientist with the topics generated by TVB together with those generated by LDA[22]. We aim to ask our social scientist the following two research questions:

- **RQ6.** Are these topics (generated by TVB) from the two communities interpretable and useful when analysing the election on Twitter?
- **RQ7.** Which group of topics (either generated by TVB or LDA) are more interpretable and useful[23]?

Next, we first describe how we present these topics to our social scientist and report the obtained answers. Then we analyse the interpretability of these topics from the two communities by using TVB in Sections 7.6.1 and 7.6.2, receptively. We analyse the divergences and similarities of these two communities in Section 7.6.3. Meanwhile, we compare the topics generated by TVB and LDA in Section 7.6.4. Note that all the topics presented in this section are the topics that we present our social scientist with.

As discussed in Section 7.5, we choose a topic model (generated by TVB) with K=70 for proClinton and a topic model (generated by TVB) with K=60 for proTrump. Rather than present our social scientist with all 130 topics across the two communities, for the purposes of visualisation and interpretation, we focus on presenting the top 18 most coherent topics from each community. We represent each topic by wordcloud[24] using its top n words, here approximately 20 words for each topic. The size of these words indicates how often they

[21] Our social scientist is an expert in political science and he is familiar with the US Election 2016 event.

[22] To ensure that the social scientist did not learn from the first experiment, we conducted the TVB and LDA comparison at a later date.

[23] We do not tell the social scientist which topic modelling approach is used to generate the topics.

[24] `https://amueller.github.io/word_cloud`

are used in the topic's discussion (i.e. $\beta_{k,i}$, introduced in Chapter 2). The blue or black colour is added for ease of interpretation. For example, Figure 7.7 shows the first 6 topics in the proClinton community, where Topic 2 is the second most interpretable/coherent topic. Similarly, we apply the classical LDA approach on the tweets of the two communities[25]. We use the same topic representation method to present our social scientist with the topics generated by LDA.

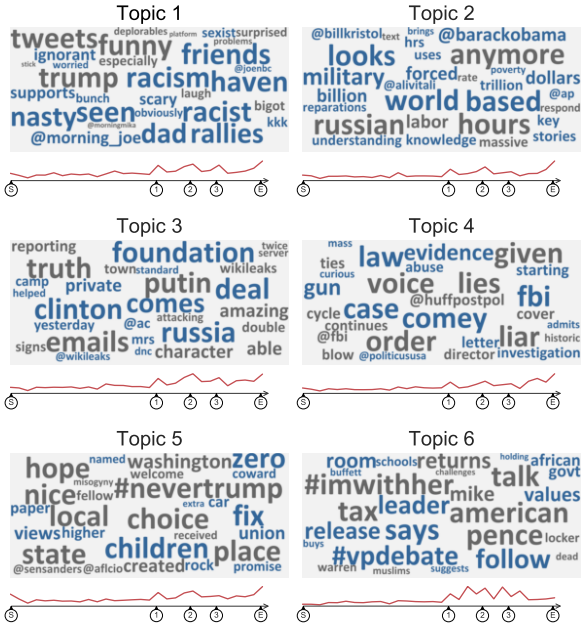

Figure 7.7: Topics extracted from proClinton (Topics 1-6) in US Election 2016 (generated by TVB).

We first presented our social scientist with the 18 most coherent topics generated by TVB from each of the two communities and asked him the first question (i.e. our sixth research question, **RQ6** in Section 7.6). Our social scientist answered that our generated topics by TVB revealed some interesting dynamics of communication on social media and the presented topics were overall interpretable and useful. Next, we presented the 18 most coherent topics generated by TVB along with the 18 most coherent topics generated by LDA for proClinton and proTrump, respectively. We asked our social scientist the second question (i.e. our seventh research question, **RQ7** in Section 7.6). Our social scientist[26] thought that there were no big differences between topics generated by TVB and LDA in

[25]We apply LDA and TVB on the same tweets (i.e. $200k$ sampled tweets for each community, described in Section 7.5.1). The setup of the classical LDA is the same as its setup in Section 5.4.2. The K in LDA is set to 70 for proClinton and 60 for proTrump, following the best K setting in Section 7.5.2

[26]Recall that we did not tell our social scientist which topic modelling approach was used to generate a given set of topics.

Figure 7.8: Topics extracted from proClinton (Topics 7-12) in US Election 2016 (generated by TVB).

terms of topical coherence. This suggests that there are no significant differences between the coherence of topics generated by TVB and LDA. However, our social scientist chose the topics generated by TVB as the more useful topics since he thought that these topics were more diverse and contained more controversial topics than those generated by LDA. Indeed, our social scientist raised very interesting points among the topics generated by TVB and LDA. We detail them in the following sections. We first analyse the topics generated from the proClinton community.

7.6.1 Analysis of proClinton Topics

In this section, we analyse the Topics 1-18 (generated by TVB) in the proClinton community in Figures 7.7, 7.8 and 7.9. Note that we include the popularity for each topic just below the word cloud in order to highlight at which moments in time that particular topic was discussed. The trend indicates the volume of tweets within the discussed topic over time. The red line in the trend represents the volume of the related tweets over our period of analysis, where the x-axis is the timeline and "S" signals the start date (01/08/2016), numbers "1", "2", and "3" denote each TV debate, and "E" represents Election Day. A spike in a trend suggests that a topic is highly discussed at that particular point in time.

Beginning with Topics 1-6 in Figure 7.7, we see a mix of topics associated more closely

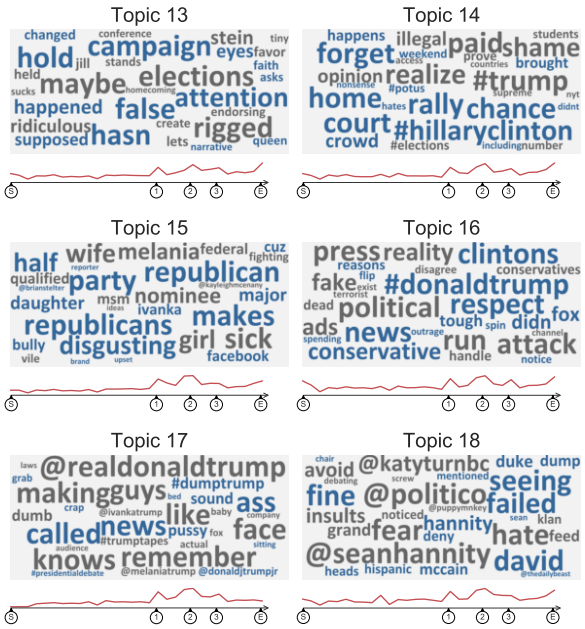

Figure 7.9: Topics extracted from proClinton (Topics 13-18) in US Election 2016 (generated by TVB).

with the Trump campaign and those more closely associated with the Clinton campaign. In Topic 1, we find a strong linkage between Trump and racism, with words such as *racism, racist, KKK, bigot, scary* included. In fact, this topic connects to several racial reviews associated with Donald Trump[27]. This antiTrump topic suggests that people who support Hillary Clinton are likely to discuss negative sides of Donald Trump. Topics 2 and 3 both have linkages to Russia and are relevant to the email scandal including words like *truth, Putin, Foundation, private* and *emails*. These two topics link to two events during the election: Russian interference[28] and Hillary Clinton email controversy[29]. Both events involve email leaking from private accounts. For example, Russian hackers accessed thousands of emails from the Chairman of Hillary Clinton's presidential campaign in March 2016 (Harding, 2016). Topic 4 continues this theme with references to the *FBI, Comey, lies/liar*, possibly linking to the FBI investigations of Hillary Clinton emails on the day just before the election date (Blake, 2017). The trends demonstrate that Topics 1 through 4 all gain momentum as the Election Day approaches. Topic 5 appears more positive than the previous ones, with words like *hope, nice, choice, children*. Topic 6 is particularly relevant to the #*vpdebate*, including *Pence* but also covering the need to release *tax returns*.

[27]Refer to `https://en.wikipedia.org/wiki/Racial_views_of_Donald_Trump`

[28]Refer to `https://en.wikipedia.org/wiki/Russian_interference_in_the_2016_United_States_elections`

[29]Refer to `https://en.wikipedia.org/wiki/Hillary_Clinton_email_controversy`

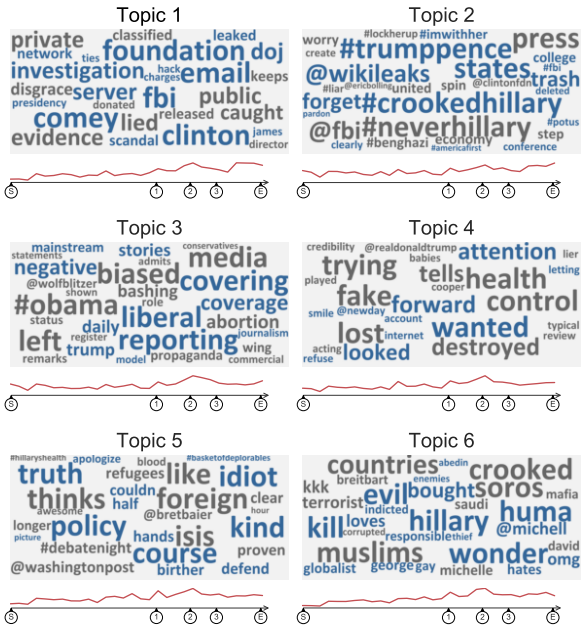

Figure 7.10: Topics extracted from proTrump (Topics 1-6) in US Election 2016 (generated by TVB).

Among the next 7 most coherent topics (Topics 7-12 in Figure 7.8), we again see a mix of topics with some pertaining more directly to Trump, and others related more to Clinton. For example, words like *sexual assault, rape, dangerous, Billy Bush* appear in Topics 7 & 8 ostensibly related to the allegations against Trump and the *access hollywood* tape. Concerns over *unfair trade, middle class* and *China* appear in Topic 9. Topics 10 through 11 have a mix of more positive words associated with the Clinton campaign such as *job, hiring* and *#ClintonKaine*, whereas Topic 12 again returns to tackling of the Trump campaign pledges with *build wall*.

Among Topics 13-18 in Figure 7.9, Topics 13 and 14 seem not to be coherent enough, however, the word *rigged* is relevant to a suspect election fraud (Wines, 2016). Topics 15 and 18 seems related to Trump, whereas Topic 13 is about the nomination of the Republican party and Topic 15 is an antiTrump discussion related to *hollywood tape* again. Topic 16 talks about *fake news*. *Sean Hannity* is mentioned in Topic 18. The reason could be that *Sean Hannity* had a claim about the healthy condition of Hillary Clinton (Fisher, 2016).

7.6.2 Analysis of proTrump Topics

In this section, we analyse the 18 most coherent topics generated by TVB in the proTrump community shown in Figures 7.10, 7.11 and 7.12. We start with the first 6 topics in Fig-

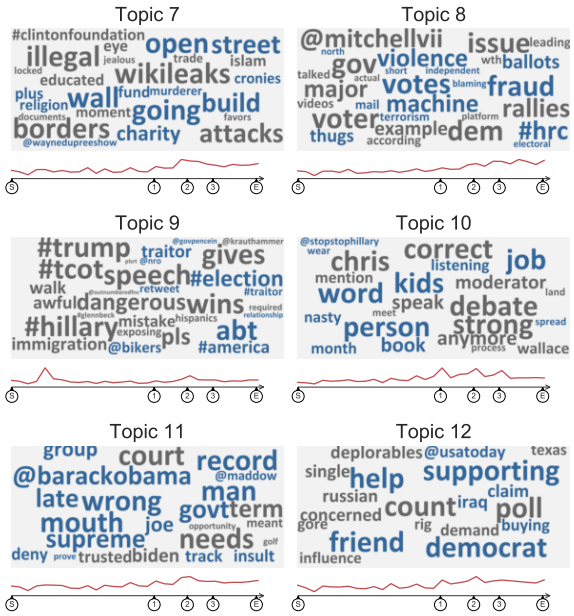

Figure 7.11: Topics extracted from proTrump (Topics 7-12) in US Election 2016 (generated by TVB).

ure 7.10. Similar to Topic 1 in the proClinton community in Figure 7.7, we find that people in the proTrump community paid attention to his opponent. The words like *foundation, email, Clinton, Comey* all appear in Topic 1, with considerable discussion from the second debate onward, and then another peak just before Election Day when Comey announced that the emails were being examined once more (Blake, 2017). Topic 2 sees a number of mentions of *#CrookedHillary* and *#NeverHillary* along with apparent trolling by the opposition with *#ImWithHer*[30] used. Topic 3 points to perceived *media bias, coverage/covering, left, propaganda, Obama*, which is associated with a raised debate about whether media was biased towards on candidates (Sides, 2016). Topics 5 and particularly 6 mention concerns over *foreign, policy, ISIS* and *muslims*. These two topics link to Trump's foreign policy. For example, Trump first proposed a *Muslims* ban in December 2015 (LoBianco, 2015).

Topics 7 through 12 in the proTrump community (shown in Figure 7.11) also provide an important lens to understand Trump support. Topic 7 invokes the *border wall* and *illegal* while also bringing in *#wikileaks* and the *#ClintonFoundation*. Words *border wall* and *illegal* connect to Trump's pledge on building a wall along the southern border (Corasaniti, 2016), which was proposed to keep *illegal* immigrants. Topic 8 turns attention to *voter fraud, machine, ballots*, pointing to a popular topic on Twitter, i.e. the election vote fraud. Topic 9

[30]Note that the font size of *#ImWithHer* in Figure 7.10 is not as big as *#CrookedHillary*, which suggests that *#ImWithHer* is not highly used in the proTrump community.

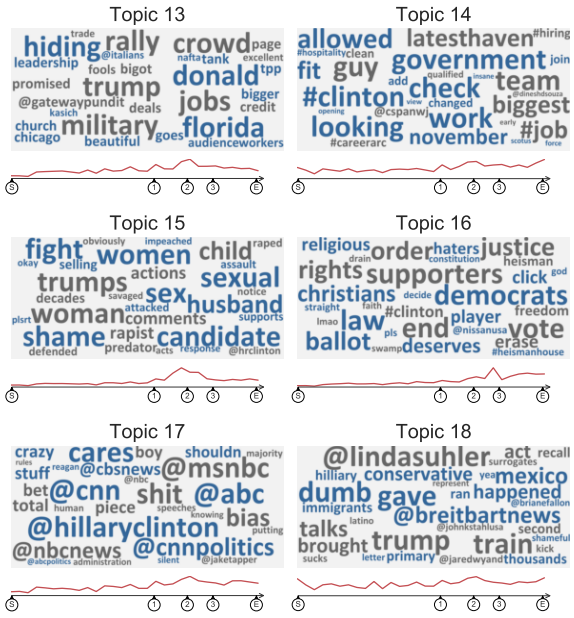

Figure 7.12: Topics extracted from proTrump (Topics 13-18) in US Election 2016 (generated by TVB).

is an example of a topic that appeared early on in our period of analysis but was relatively quiet thereafter, touching on several themes including *immigration* and the candidate names and general words like *election, America*. Topic 10 has particular relevance to the debates and debate moderation (e.g. *Chris Wallace, debate*). Topic 11 links largely to the Obama administration and concerns over a Supreme Court appointment (e.g. *Biden, record, Supreme Court*) and includes apparent trolling of the former president through *@barackobama*. Topic 12 represents another mix of terms such as *Democrat, friend, Deplorables*.

Among Topics 13-18 in Figure 7.12, we see *job*-related topics in the proTrump community, i.e. Topics 13 and 14, which are similar to Topics 10 and 11 in the proClinton community. The *women* discussion is raised again in the proTrump community in Topic 15. Topic 16 connects to *christians vote*. Again, the *media bias* is mentioned in Topic 17 where media outlets are listed. Topic 18 raises the topic of *mexico immigrants* similar to Topic 12 in the proClinton community.

7.6.3 Analysis of Topics across both Communities

Among the topics discussed across the two communities, we find that the supporters of one candidate are likely to discuss topics that oppose another candidate. For example, Topics 1 and 8 in the proTrump community and Topics 1 and 2 in the proClinton community. The

controversial topics are popular among both communities, such as the topic of email leaking and building the border wall. Our social scientist mentioned that he did not observe a lot of policy-related topics. Indeed, we only find two topics (i.e. Topics 5 and 6 in Figure 7.10) that are related to Trump's foreign policy. This might be because the volume of the tweets related to the election policy is not as big as the tweets related to the observed topics and therefore our TVB approach has difficulties when identifying the topics with a small number of tweets. We also observe that there are connections between the two communities. For example, both communities discussed the topics of *email leaking*, *mexico immigrants wall* and *jobs*. This might be because that such topics raised a big concern and Twitter users in both communities tended to comment on these topics. On the other hand, one big divergence is that the two communities use different words for similar topics. For example, both communities discussed *email leaking*, i.e. Topic 3 in Figure 7.7 in the proClinton community and Topic 1 in Figure 7.10 in the proTrump community. However, in the proClinton community, the topic of *email leaking* has words, such as *putin*, *russia* and *truth*, which are different from the same topic in the proTrump community, where *FBI*, *investigate* and *evidence* were instead highly used. This suggests that the supporters in the proClinton community are likely to link the Russian intervention with *email leaking* during the election while the supporters in the proTrump community tended to discuss the investigation of Hillary Clinton *email leaking*.

Thus far, we have shown the topics generated by the TVB approach from the two identified communities. We have also analysed the main similarities and divergences among the two communities. Other social science tasks can be conducted by using the discussed topics. For example, social scientists can generate community-related topics over time, which can allow them to examine the dynamics of the topics and therefore can help to study the dynamics of the two communities during the election. Moreover, they can examine how many Twitter users were involved in a topic discussion, which can be then used to identify which topic was popular in a given community. Alternatively, a graph model (e.g. as used by Grčar et al., 2017) can be applied to identify the influential Twitter users in a topic of a community. In Chapter 6, we investigated a user community classifier that associates Twitter users into four communities: Academics, Media, Business Elites, and Politics (see Section 6.2.2.1). The trained classifier can be similarly applied to categorise users in the US Election 2016 dataset (the election-related tweets) into the four communities. It will be indeed interesting to see what topics were highly discussed in these four communities. In fact, we did apply our TVB approach to extract topics from these four communities shown in Figures B.1-B.4 in the Appendix, where we also analyse several popular topics discussed in the four communities. All these possible social science studies can be conducted to analyse the election by using our Twitter user community classification approaches and our time-sensitive topic modelling approach.

7.6.4 Comparison to the Classical LDA Approach

We also present our social scientist with topics generated by TVB together with those generated by LDA in order to identify which approach generates more interpretable and useful topics. Recall that the social scientist did not know which topic modelling approaches generated a given set of topics. In this section, we compare the 6 most coherent topics generated by TVB (shown in Figures 7.7 and 7.10) to the 6 most coherent topics generated by LDA[31] (shown in Figures 7.13 and 7.14) discussed in each of the two communities. This allows us to examine the differences of topics generated by using TVB and LDA.

Figure 7.13: Topics extracted from the proClinton community using the classical LDA approach.

The first comment we obtain from our social scientist is that the topics generated by TVB reveal more controversial topics. On the other hand, the topics generated by LDA are more standard topics that are commonly discussed on Twitter. For example, the most coherent topic, Topic 1 (generated by LDA, see Figure 7.14) in proTrump is about the states that were contested, e.g. *florida*, *michigan* and *#northcarolina*. Topic 1 in proClinton (generated by LDA, see Figure 7.14) is about *hrc* (Hillary Rodham Clinton). On the other hand, among the topics generated by TVB, we observe a controversial topic, *racist*, in proClinton as the most coherent topic and another controversial topic about *FBI* and *investigation* in proTrump as the most coherent topic (see Figures 7.7 and 7.10). The controversial topics happened in specific time periods, e.g. the topic of the FBI investigation of Hillary Clinton's emails and the topic of *Hollywood tape*. The topical trends of the controversial topics usually have peaks indicating when the topics were highly discussed. Meanwhile, a common topic

[31]The other 12 topics generated by the classical LDA approach are shown in Figures B.5-B.8 in the Appendix.

Figure 7.14: Topics extracted from the proTrump community using the classical LDA approach.

can be discussed all the time and has a flat topical trend. Our TVB approach extracts topics by considering the time dimension of the topics. Since the topical trends of the controversial topics are very different from the other topics, TVB appears to better identify these topics compared to LDA.

Second, our social scientist commented that scholars had interests in the dynamics of conversation, e.g. the breaking news about the FBI investigation on Hillary Clinton's emails just before the election date. Such a topic could change the result of the election. This topic is identified by TVB and appeared as the most coherent topics in proTrump (see Topic 1 in Figure 7.10). This topic can be seen as an anti-Clinton topic and it was highly discussed before the election date. On the other hand, although LDA identifies this topic as the second most coherent, this topic (Topic 2 in Figure 7.14) does not contain the name of the key people, e.g. James Comey (the former director of FBI). It suggests that Topic 2 in Figure 7.14 might not be necessarily related to the long discussion about the email scandal of Hillary Clinton. It seems that our TVB approach can better deal with the dynamics of conversation on Twitter.

Third, our social scientist thought that the topics generated by TVB are more diverse. For example, the *email* discussion appears in Topic 3 in proClinton and appears in Topic 1 in proTrump when generated by TVB (see Figures 7.10 and 7.7). On the other hand, this discussion shows up in Topics 4 and 6 in proClinton and Topics 2 and 6 in proTrump when generated by LDA (see Figures 7.13 and 7.14). This might because that the *email* discussion mixed with other topics and thereby appeared in different topics in the LDA model. TVB generates less mixed topics than LDA (see Section 5.5.3) and therefore could generate more different topics, which might be the reason its topics are more diverse.

Since our TVB approach appears to generate more interesting controversial topics, seems to better capture the dynamics of Twitter conversation and have more diverse topics, our social scientist chose the topics generated by TVB as more useful than those generated by LDA to analyse the US Election 2016 event.

7.7 Conclusions

In this chapter, we applied our proposed TVB approach, our topic coherence metrics, and our user community classification approaches to analyse US Election 2016 on Twitter. We verified the generalisation of the results obtained from the previous chapters in a Twitter dataset of a major political event, i.e. US Election 2016. We showed that our proposed hashtag labelling approach can be applied to generate effective ground-truth data for training a user community classifier. By using the generated ground-truth data, we demonstrated that our TBNB approach can more effectively identify the community affiliations of Twitter users than commonly used classifiers in our US election Twitter dataset, such as NB and SVM (see Tables 7.4 and 7.5). We also conducted a user study to obtain human ground-truth labels for 100 Twitter users sampled from our unlabelled dataset. We showed that our TBNB classifier also had a higher agreement with the human assessors compared to the other commonly used classifiers on the unlabelled dataset (see Table 7.5). To identify what topics these two communities discussed during the election, we applied our proposed TVB approach to extract coherent topics from the two communities by considering when tweets were posted (c.f. Section 7.5). We used our proposed Twitter topic coherence metrics (the WE-based metric and the `coherence at n` metric) to evaluate topic models and to select coherent topics. We also compared the topics generated by using our TVB approach to those generated by using the classical LDA approach. We confirmed with a social scientist that our TVB approach can generate interpretable topics. In particular, topics generated by using TVB can be more useful when analysing the election since these topics appear to be more diverse and contain more controversial topics, of interest to social scientists, compared to those generated by the classical LDA approach. Finally, we analysed the similarities and divergences among the proTrump and proClinton communities using the generated community-related topics (c.f. Section 7.6). In summary, in this chapter, we have demonstrated the effectiveness of our proposed approaches in assisting social scientists when analysing a political event on Twitter.

Chapter 8

Conclusions and Future Work

8.1 Conclusions and Contributions

In this thesis, we proposed a series of approaches to understand 'who' said 'what' and 'when' during a political event on Twitter. We argued that identifying the 'who' (i.e. communities) can be conducted through an automatic user community classification approach, while the 'what' (i.e. the discussed topics) can be addressed through a tailored topic modelling approach that integrates the time dimension (i.e. 'when') of tweets. For the automatic classification of users into communities, we first proposed two automatic ground-truth generation approaches to train and develop user community classifiers (see Chapter 6). To effectively classify the community affiliations of Twitter users, we proposed a Topic-based Naive Bayes (TBNB) classification approach in Chapter 6. Our TBNB approach classified Twitter users by considering both their discussed topics and the used words in their tweets. We showed that our TBNB approach was promising, often outperforming baseline classifiers such as Naive Bayes and Decision Trees across two Twitter datasets (see Chapter 6). To extract what topics were discussed on Twitter during a political event, we proposed an effective time-sensitive topic modelling approach in Chapter 5. Our time-sensitive topic modelling approach integrated the time dimension of tweets and thus generated topics with a higher coherence compared to the other topic modelling approaches, such as the classical LDA topic modelling approach. In order to evaluate the coherence of the generated topics, we proposed a Twitter coherence metric based on word embedding (T-WE, see Table 4.6) that was effectively trained using a Twitter background dataset (see Chapter 4). We demonstrated that our proposed metric can better align with human judgements than other baseline met-

rics, such as the W-PMI metric based on the pointwise mutual information. To show the generalisation of our approaches, we applied our approaches on a collection of US Election 2016 tweets in order to identify which community a Twitter user belongs to and what topics they have discussed on Twitter during this election (see Chapter 7). We concluded that our approaches can indeed effectively identify the communities and extract coherent topics from tweets thereby assisting social scientists to study this particular major political event. Overall, our experiments showed that our proposed approaches permit a social scientist to understand the connections and dynamics of communities on Twitter.

In the remainder of this chapter, we first summarise the contributions of this thesis in Section 8.1.1 followed by the main achievements and conclusions of this thesis presented in Section 8.1.2. We discuss some future directions in the field of both computing science and social science in Section 8.2. Finally, we present our closing remarks in Section 8.3.

8.1.1 Contributions

The main contributions of this thesis are as follows:

- In Chapter 4, we proposed four types of topic coherence metrics (see Table 4.1) that automatically evaluate the coherence of topics generated using topic modelling approaches. In addition, we examined the performance of eight existing topic coherence metrics, which use techniques such as pointwise mutual information. Moreover, to increase the coverage of words occurring in tweets, we proposed to use a Twitter background dataset as an external resource to obtain effective word embeddings (see Section 4.3). We conducted a large-scale user study (168 users) to obtain coherence judgements of topics from humans and then evaluated our proposed coherence metrics together with the existing coherence metrics using the obtained human judgements (see Section 4.4). While the topic coherence metrics evaluate a single topic, we also proposed an approach for calculating the global coherence of a topic model containing many topics (see Section 4.6). Inspired by the `precision at n` information retrieval metric, we proposed the `coherence at n` metric to evaluate the coherence of a topic model and compared our metric to the commonly used average coherence score. To validate the usefulness of the `coherence at n` metric, we conducted a user study (52 users) to obtain human judgements on topical preferences.

- In Chapter 5, we proposed a time-sensitive topic modelling approach for Twitter data. We studied LDA approaches based on the traditional Gibbs sampling and the more recent Variational Bayesian inference (VB) in terms of generating coherent topics from

Twitter data. We integrated the time dimension of tweets in the VB-based LDA approach (called TVB) in Section 5.2. To evaluate our proposed TVB approach, we conducted experiments using two real-world Twitter datasets and we evaluated our proposed TVB approach using our proposed T-WE Twitter topic coherence metric and the `coherence at n` metric. In addition, we evaluated how likely a topic modelling approach generates mixed topics that combine multiple themes using our proposed topic mixing degree (`MD`, introduced in Section 5.4.3.2) metric. To evaluate the proposed `MD` metric, we conducted a user study, where 8 expert users were asked to identify all the mixed topics from a topic model. Finally, to demonstrate the effectiveness of our TVB approach when estimating the topical trends, we computed the errors between the real topic trends and the estimated topic trends (see Figure 5.6). We demonstrated that our TVB approach has clear advantages in generating coherent and less mixed topics.

- In Chapter 6, we proposed two automatic ground-truth data generation approaches and a user community classification approach for identifying the community affiliations of Twitter users. We proposed to use hashtags to automatically label the communities in a referendum or an election. We denoted this first approach as the hashtag labelling approach (see Section 6.2.1). We also proposed to use the DBpedia entities to label the communities in terms of their users' professions. We denoted this second approach as the DBpedia labelling approach (see Section 6.2.2). We evaluated the proposed hashtag labelling approach using a Twitter followee network while we evaluated the proposed DBpedia labelling approach using a user study (124 users), where we asked humans to label the community affiliations of Twitter users. We proposed a novel Topic-based Naive Bayes (TBNB) approach, which identified the communities of Twitter users using both their posted words and the topics these communities discussed (see Section 6.3). We conducted experiments using two Twitter datasets generated using each of the hashtag labelling and DBpedia labelling approaches. We evaluated our proposed TBNB approach in comparison to several other baseline classifiers that are commonly used in the literature (see Section 6.4).

- In Chapter 7, we contributed an application towards analysing US Election 2016 on Twitter using our proposed approaches, which also demonstrated the generalisation of our previously obtained results to another large election-related dataset. In particular, we applied the hashtag labelling approach to obtain an effective ground-truth dataset containing two communities of Twitter users in favour of the two presidential candidates in the election. We applied our TBNB approach to classify our election-related tweets (3.6 million) into the two communities (see Section 7.4). To evaluate the perfor-

mance of our TBNB classifier in identifying the Twitter users' communities, we also conducted a user study (31 users) to obtain the community labels of 100 Twitter users sampled from an unlabelled dataset (see Section 7.3). We then applied our proposed time-sensitive topic modelling approach to obtain the community-related topics. To choose the most coherent topics, we applied our T-WE Twitter topic coherence metric (see Section 7.5.2). Finally, with the help of a social scientist, we analysed these generated community-related topics to examine the dynamics between the two communities during the election (see Section 7.6).

8.1.2 Conclusions

In this section, we summarise the main conclusions and achievements of this thesis. Taken together, these conclusions validate our thesis statement proposed in Section 1.3.

- ***Effectiveness of Twitter Topic Coherence Metrics for Evaluating the Coherence of Latent Topics:*** In Chapter 4, we showed that our proposed word embedding (WE)-based metrics trained using a Twitter background dataset had a consistently high-level agreement with human judgements in terms of agreement and $Kappa$ scores across two real-world Twitter datasets (see Figures 4.5 & 4.6). We also found that our proposed WE-based metrics can accurately identify the coherence performance differences among three topic modelling approaches,in comparison with human judgements, across the two used Twitter datasets (see Tables 4.7 & 4.8). Hence, we concluded that our proposed WE-based coherence metrics were consistently effective across two different Twitter datasets, often outperforming the performance of the other baseline metrics, such as W-LSA, W-PMI and W-WE (see Table 4.6).

- ***Effectiveness of the `Coherence at n` Metric for Evaluating the Coherence of Topic Models:*** In Chapter 4, we proposed a `coherence at n` metric to evaluate the coherence of a topic model containing K topics. We conducted a large-scale experiment on two real-world Twitter datasets. We found that all used topic modelling approaches generated topics with a higher coherence when the number of topics K increased (see Figure 4.7 and Table 4.10). We showed that our `coherence at n` metric can capture the changes in the coherence of topics when K increases while the average coherence score cannot identify such changes (see Figure 4.7). In our user study, we demonstrated that our `coherence at n` metric can effectively evaluate the coherence of the top-ranked topics in a manner that is aligned with human assessors. Therefore, we concluded that our `coherence at n` metric was effective when evaluating topic models.

- ***Effectiveness of a Time-sensitive Topic Modelling Approach for Twitter data:*** In Chapter 5, we proposed a time-sensitive topic modelling (TVB) approach to generate coherent topics from Twitter data. We compared our approaches with four other topic modelling approaches, i.e. the classical Gibbs sampling, the classical VB approach (VB), the Twitter LDA (TLDA) and the topic over time LDA (TOT). We showed that, when integrating the time dimension, our TVB approach indeed generated topics that are significantly more coherent than the VB approach across the two Twitter datasets (see Tables 5.3, 5.7 and 5.8), demonstrating the usefulness and effectiveness of the time dimension. We showed that our TVB approach also generated topics that are significantly less mixed than those generated by TLDA (see Tables 5.3, 5.7 and 5.8). Moreover, we demonstrated that our TVB approach can significantly more accurately estimate the trends of topics (see Figure 5.6) compared to TOT. Hence, we concluded that our time-sensitive topic modelling approach was overall promising and effective when extracting topics from Twitter data.

- ***Effectiveness of Ground-truth Generation approaches and a Topic-based Naive Bayesian Classification approach for Twitter User Community Classification:*** In Chapter 6, we proposed two automatic ground-truth generation approaches and a TBNB approach for classifying Twitter users into communities. The ground-truth labelling approaches included the hashtag labelling and DBpedia labelling approaches. We showed that our hashtag labelling approach had a high agreement with the followee network verification method (see Table 6.1), which demonstrate that our hashtag labelling approach was effective when generating ground-truth data. Through a user study, we also showed that our DBpedia labelling approach had a good-level agreement with human judgements (see Table 6.6). To effectively classify Twitter users into communities, we proposed the TBNB approach. In order to evaluate our TBNB approach, we conducted experiments on both the IndyRef and the DBpedia community datasets (i.e. generated using our two proposed ground-truth generation approaches). We showed that our TBNB approach was overall promising, often significantly outperforming several commonly used classifiers, such as Naive Bayes and Decision Trees (see Tables 6.10 and 6.12) in terms of micro-F1 across both used datasets. We concluded that the proposed ground-truth labelling approaches permit to effectively train a TBNB classifier that accurately identifies the community affiliations of users on Twitter.

- ***Effectiveness of our Proposed Approaches in Analysing the US Election 2016 event:*** In Chapter 7, we applied our proposed approaches to analyse US Election 2016 on Twitter. Through a user study, we demonstrated that our hashtag labelling approach

generated effective ground-truth data for training a user community classifier (see Table 7.5). We also showed that, using the generated ground-truth data, our TBNB classifier can more accurately classify users into two communities than the baseline classifiers that are commonly used in the literature (see Table 7.4). With the input of a social scientist, we showed that our TVB time-sensitive topic modelling approach can leverage the time dimension of tweets to effectively extract interpretable topics from the two classified communities compared to the classical LDA (see Figures 7.7-7.14). We also showed that our Twitter coherence metric can be used to evaluate the topic models (see Figure 7.6) and choose the most interpretable topics (presented in Figures 7.7-7.12). Overall, our results in Chapter 7 do appear to support the generalisation of our previous conclusions to a large dataset of tweets related to a major political event.

- ***Validating our Thesis Statement:*** The statement of this thesis is that we can use a series of approaches to understand 'who' said 'what' and 'when' during a political event on social media networks, i.e. Twitter.

    1. We claimed that identifying the 'who' benefits from an automatic user community classification approach. We argue that we have validated this claim in Chapter 6 where we showed that the hashtags and mentioned entities can be used to generate effective ground-truth data for user community classification. At the same time, the discussed topics in tweets can help to more effectively classify Twitter users' community affiliations using our proposed TBNB approach compared to the commonly used classifiers (see Tables 6.10 and 6.12).
    2. We claimed that the 'what' can be addressed by modelling the topics of conversations while taking into account the importance of the time dimension (i.e. 'when') on Twitter. We argue that we have validated this claim in Chapter 5 where we demonstrated that our proposed time-sensitive approach, which integrated the time dimension of tweets, can generate topics with a higher coherence and less mixed topics compared to several existing baseline topic modelling approaches (see Tables 5.3, 5.7 and 5.8).
    3. We claimed that the coherence of the generated topics can be effectively evaluated using word embeddings that are trained using a Twitter background dataset. We argue that we have validated this claim in Chapter 4 where we showed that our proposed word embedding-based coherence metric (trained using Twitter background dataset) can more effectively capture the semantic similarities of words in tweets and thus can more effectively evaluate the coherence of topics compared

to several existing coherence metrics, such as the pointwise mutual information-based metric (see Figures 4.5 & 4.6 and Tables 4.7 & 4.8).

Finally, to show the generalisation of our proposed approaches towards analysing a political event on Twitter, we demonstrated an application in Chapter 8 where we showed that our proposed approaches can effectively identify communities and extract coherent topics from tweets related to US Election 2016. Therefore, we argue that we have validated the statement of this thesis.

8.2 Directions for Future Work

In this section, we discuss possible directions for future research. We split our future directions into two parts. In the first part, we discuss future work in the field of computing science. This includes more possible computing science approaches (based on our proposed approaches) that could assist social scientists. In the second part, we present future research directions in the field of social science. We discuss possible social science studies that can be conducted using our approaches.

Research Direction in Computing Science

- ***Integrating communities in Topic Modelling:*** In Chapter 5, we have introduced a time-sensitive topic modelling approach tailored to Twitter by integrating the time dimension of tweets. A tweet is associated with a timestamp while a Twitter user can belong to a community, such as business elites and academics (see Chapter 6). Such community labels can be classified using the trained classifier in Chapter 6. Therefore, the communities can be also integrated in topic modelling similarly to the time dimension. It will be interesting to identify whether the integration of the communities into the topic modelling process can improve the coherence of the generated topics.

- ***Study the Topic Coverage of the Topic Modelling Approaches:*** In this thesis, we focused on generating topics that are easier for a human to understand. It is also interesting to study how likely the real topics discussed in a corpus can all be extracted by a topic modelling approach, i.e. whether the generated topics can cover all the real topics in a corpus (topic coverage). In Chapter 4, we showed that the coherence of topics can be improved by setting a larger number of topics. Similarly, it is likely that a larger number of topics can help to generate topics that cover more real topics.

- ***Study of the Evolution of Topics on Twitter:*** In Chapter 5, we improved the coherence of topics by integrating the time dimension of tweets in our time-sensitive topic

modelling approach. On the other hand, the time dimension of tweets can help to examine how a topic changes over time, i.e. the evolution of topics. Such topic evolutions are popular in the field of social science (e.g. in Mascaro et al., 2012; Liu et al., 2013). Blei and Lafferty (2006) proposed a dynamic topic modelling approach, where the topic word distributions were adjusted for each time interval. It is important to investigate how this approach performs on Twitter data, e.g. how it performs for a large number of tweets, whether the topic word distributions over time can capture the real changes of topics or how efficient this approach is. In fact, we have recently started working towards this research direction, by investigating how to predict whether a topic will burst in the future using the time dimension of tweets (Fang et al., 2018b).

- ***Further Study of the Quality of Topics:*** In Chapter 4, we demonstrated that the coherence of the generated topics can be evaluated by the topic coherence metrics. Such metrics evaluate how likely a topic can be interpreted by humans. However, there can be duplicated topics (i.e. topics that are under the same topic theme) among the generated topics. Such duplicated topics can cause extra time for users to interpret especially when there are a large number of generated topics from tweets. Therefore, it can be important to study the metrics to measure the similarities of generated topics from a given topic model, so as to identify the duplicated topics.

- ***Classifying Twitter Users using Deep Learning Techniques:*** In Chapter 7, we proposed two ground-truth generation approaches and a TBNB approach for Twitter user community classification. Although we obtained 89% accuracy (see Table 6.4.4) for the Scottish Independence Referendum 2014 (two-class classification), the accuracy on the DBpedia (four-class) community classification was not that high, around 65% (see Table 6.4.5). Due to the emergence of deep learning techniques, it is worth investigating whether these approaches can further improve the performance of the Twitter user community classification task. For example, Kim (2014) proposed convolutional neural networks for classifying short sentences, which we believe can be applied to Twitter user community classification.

Research Direction in Social Science

- ***Study the Changes of Voting Preferences in an Election:*** Political campaigns can impact voters' decisions during an election (Cantrell, 1992). Polling has been used widely to monitor the changes of voters' decisions (e.g. in Hillygus, 2011). Since there are more and more election-related topics discussed on Twitter, it is possible to analyse the discussed topics, to identify the communities of Twitter users and therefore

to examine how voters change their decisions during an election. More importantly, by classifying Twitter users into communities and extracting topics these communities discussed over time, it will be interesting to examine the evolution of the topics and the changes of voting preferences over time, during a political event.

- ***Study the Influence of Twitter Users within Topics and Communities:*** The existing work often used graph models to measure the influence of Twitter users (e.g. in Grčar et al., 2017; Habel et al., 2018), where the user followee network was used to construct the graph models. These work investigated users' influences in a political event or among a group of Twitter users. In Chapter 6, we showed that the discussed topics can reflect the political orientations of Twitter users. Therefore, it would be interesting to study which Twitter users have an influence in a given topic and who are leading the topic, which might further help to understand how Twitter users vote.

8.3 Closing Remarks

In this thesis, we have addressed a challenging task, namely building a series of approaches to assist social scientists to understand 'who' said 'what' and 'when' within a political event on Twitter. We identified the 'who' by using the proposed Twitter user community classification approaches and addressed the 'what' by using the proposed time-sensitive topic modelling approach that takes into account the importance of the time dimension of tweets (i.e. 'when').

We have argued that an effective Twitter topic coherence metric can be achieved by using word embeddings trained using a Twitter background dataset. We showed that this proposed metric aligned with human preferences when assessing the coherence of topics. We have argued that an effective time-sensitive topic modelling approach can be achieved by considering the importance of the time dimension on Twitter. We showed that our proposed time-sensitive topic modelling approach can generate topics with a higher coherence and less mixed topics on Twitter data compared to the other commonly used topic modelling approaches. We demonstrated that the mentioned entities and hashtags on Twitter can be used to automatically generate effective ground-truth data and that the discussed topics can be used to effectively identify the community affiliations of Twitter users. Finally, to show the generalisation of our approaches, we applied our approaches to successfully analyse US Election 2016 on Twitter.

In this thesis, we have made progress to assist social scientists towards analysing political events on Twitter. However, there are many other challenging and interesting tasks (including those listed as future directions in Section 8.2). Recently, social scientists have

started to see the benefit of using Twitter to track and analyse political events, such as the monitoring of election campaigns (Enli, 2017) or the identification of incidents during elections (Yang et al., 2018). Therefore, it has become increasingly important to provide easy-to-use tools for social scientists to analyse these events on Twitter. At the same time, computing science approaches, such as the approaches proposed in this thesis, play an important role to process, model and leverage social media data. We expect that applying computing science approaches for social science studies will continue to be an important field in future research.

Appendix A

Tables

Table A.1: The symbols used in Chapters 2, 3 and 4.

Symbol	Description
K	The total number of topics.
k	The index of a topic.
N	The size of vocabulary. n is the index of a word.
D	The number of documents in a corpus. d is the index of a document.
U	The total number of users in a corpus.
u	The index of a user.
$\mathbf{W}$	All the documents in a corpus.
$\vec{w}_d$	The the d-th document .
N_d	The number of words in $\vec{w}_d$.
i	The index of a term in a document or a corpus.
$w_{d,i}$	The i-th word in the d-th document .
b	A bi-term.
$t_{d,i}$	The timestamp of the i-th word in the d-th document.
$z_{d,i}$	The topic assignment of $w_{d,i}$.
θ_d	The topic distribution of the d-th document.
α_d	The hyperparameter of θ_d.
γ_d	The variational hyperparameter of θ_d.
β_k	The term distribution of topic k.
η_k	The hyperparameter of β_k.
τ_k	The time distribution of topic k.
ρ_k^1/ρ_k^2	The hyperparameters of τ_k.
λ_k	The variational hyperparameter of β_k.
$\phi_{d,i,k}$	The topic distribution of $w_{d,i}$.
βuni	The uniform topic distribution.
β_{vac}	The vacuous topic distribution.
ϑ_{bg}	The background topic distribution.

Table A.2: The combinations of Predicate & Object for the ACA and MDA communities.

Community	Combination Predicate & Object
ACA	subject:Category & Academic_administration subject:Category & Science_occupations subject:Category & Research subject:Category & University_and_college_people subject:Category & Academic_ranks 22-rdf-syntax-ns#type & Institution108053576 22-rdf-syntax-ns#type & EducationalInstitution108276342 22-rdf-syntax-ns#type & EducationalInstitution 22-rdf-syntax-ns#type & University 22-rdf-syntax-ns#type & EducationalOrganization 22-rdf-syntax-ns#type & CollegeOrUniversity 22-rdf-syntax-ns#type & EducationalOrganization 22-rdf-syntax-ns#type & University
MDA	subject:Category & Journalism_occupations subject:Category & Media_occupations subject:Category & Broadcasting_occupations subject:Category & Journalists subject:Category & Journalism subject:Category & Broadcast_news_analysts 22-rdf-syntax-ns#type & Medium106254669 22-rdf-syntax-ns#type & Newspaper106267145 22-rdf-syntax-ns#type & Press106263369 22-rdf-syntax-ns#type & PrintMedia106263609 22-rdf-syntax-ns#type & BroadcastingStation102903405 22-rdf-syntax-ns#type & RadioStation104044119 22-rdf-syntax-ns#type & NewsAgency108355075 22-rdf-syntax-ns#type & TelevisionStation 22-rdf-syntax-ns#type & Newspaper 22-rdf-syntax-ns#type & Broadcaster 22-rdf-syntax-ns#type & TelevisionStation

Table A.3: The combinations of Predicate & Object for the BE and PLT communities.

Community	Combination Predicate & Object
BE	subject:Category & Business_occupations
	22-rdf-syntax-ns#type & Enterprise108056231
	22-rdf-syntax-ns#type & Company108058098
	22-rdf-syntax-ns#type & Business108061042
	22-rdf-syntax-ns#type & Administrator109770949
	22-rdf-syntax-ns#type & Executive110069645
	22-rdf-syntax-ns#type & Business108061042
	22-rdf-syntax-ns#type & BusinessPerson
PLT	subject:Category & Legislators
	subject:Category & Parliamentary_titles
	subject:Category & Government_occupations
	subject:Category & Positions_of_authority
	subject:Category & Political_occupations
	subject:Category & Political_staffers
	subject:Category & Organizational_structure_of_political_parties
	22-rdf-syntax-ns#type & Legislators
	22-rdf-syntax-ns#type & Politician110451263
	22-rdf-syntax-ns#type & GovernmentOccupations
	22-rdf-syntax-ns#type & Politician110450303
	22-rdf-syntax-ns#type & Legislature108163273
	22-rdf-syntax-ns#type & Legislature
	22-rdf-syntax-ns#type & PoliticalParty
	22-rdf-syntax-ns#type & GovernmentalOrganization

Experiments on the Refined Baseline Dataset

We randomly select 10% of the Twitter users from the refined baseline training dataset (see Table 6.8) as a test dataset to evaluate the performance of the classifiers trained using the refined baseline training dataset (the selected Twitter users in the test dataset are removed) and the DBpedia training dataset, reported in Table A.4 (a) and (b), respectively. As discussed in Section 6.4.1, the Twitter users in the refined baseline training dataset are not general Twitter users and they are easier for the classifiers to categorise into communities. For example, the accuracy score of MLP_{RDB} (trained using the refined baseline training dataset) is 0.89. This is because the Twitter users in the public lists share similar interests and use similar words in their tweets. On the other hand, we also observe that the classifiers (i.e. classifiers with "*DBD*" as subscript in Table A.4 (b)) trained using the DBpedia training dataset perform reasonably well, although they are not as good as the classifiers (i.e. classifiers with "*RBD*" as subscript in Table A.4 (a)) trained using the refined baseline training dataset. For example, NB_{DBD} has an accuracy F1 score of 0.69. However, the performance of NB_{DBD} is better than the best achieved performance (an accuracy of 0.65) shown in Table 6.11. This shows that the results of the classifiers trained using the DBpedia training dataset (see Section 6.4.5) generalise to this new test dataset.

Table A.4: Additional user community classification results for Chapter 6.

(a) Classifiers trained using the **refined baseline training dataset**.

		ACA	MDA	BE	PLT	Accuracy
RDN	F1	0.233	0.279	0.141	0.272	0.231
	Precision	0.208	0.290	0.163	0.258	
	Recall	0.264	0.268	0.125	0.288	
DT_{RBD}	F1	0.530	0.564	0.704	0.452	0.563
	Precision	0.500	0.578	0.741	0.444	
	Recall	0.566	0.552	0.671	0.461	
NB_{RBD}	F1	0.851	0.879	0.901	0.783	0.854
	Precision	0.836	0.948	0.820	0.844	
	Recall	0.867	0.820	1.0	0.731	
SVM_{RBD}	F1	878	0.872	0.939	0.800	0.872
	Precision	0.870	0.878	0.911	0.833	
	Recall	0.887	0.865	0.968	0.769	
MLP_{RBD}	F1	0.880	0.902	0.945	0.831	**0.890**
	Precision	0.857	0.909	0.938	0.857	
	Recall	0.905	0.895	0.953	0.807	

(b) Classifiers trained using the **DBpedia training dataset**.

		ACA	MDA	BE	PLT	Accuracy
RDN	F1	0.192	0.310	0.200	0.218	0.233
	Precision	0.192	0.263	0.226	0.236	
	Recall	0.192	0.377	0.179	0.203	
DT_{DBD}	F1	0.459	0.314	0.451	0.483	0.535
	Precision	0.400	0.279	0.491	0.560	
	Recall	0.538	0.358	0.417	0.359	
NB_{DBD}	F1	0.707	0.530	0.793	0.723	**0.6991**
	Precision	0.655	0.577	0.812	0.712	
	Recall	0.769	0.490	0.776	0.734	
SVM_{DBD}	F1	0.648	0.591	0.713	0.783	0.686
	Precision	0.625	0.548	0.741	0.839	
	Recall	0.673	0.641	0.686	0.734	
MLP_{DBD}	F1	0.631	0.571	0.682	0.783	0.669
	Precision	0.580	0.542	0.728	0.839	
	Recall	0.692	0.603	0.641	0.734	

Appendix B

Figures

We apply the trained classifier (i.e. trained using the DBpedia training dataset) in Section 6.4 to categorise the Twitter users in the US Election 2016 unlabelled dataset (i.e. 264k Twitter users with 3.5m posted tweets, see Table 7.6) into four communities: Academics (ACA), Media (MDA), Business Elites (BE), and Politics (PLT) (see Section 6.2.2.1). This leads to 26k, 37k, 32k and 168k users belonging to the ACA, MDA, BE and PLT communities, respectively. We observe that the PLT community contains a lot more users than the other three communities. We randomly sample $20k$ tweets from each of the four communities and apply our time-sensitive topic modelling (TVB) approach to extract 60 topics from each of the four communities (the setup of TVB is the same as its setup in Section 7.5). We present the 6 most coherent topics of the four communities in Figures B.1-B.4, respectively.

Figure B.1: The 6 most coherent topics extracted from the ACA community in US Election 2016.

Figure B.1 shows the 6 most coherent topics from the ACA community. We can see several topics that are related to academia. For example, Topic 1 has words such as *education* and *presidents*, which possibly link to some *education* discussions during the election. Similarly, we see words such as *student*, *afford*, *pay* and *debates* in Topic 2, which can link to the discussions of the university tution fees during the TV debates. We observe that Topic 5 is likely to be a proClinton topic since it has the hashtag *#imwithher*. Topic 6 is related to *#job* and *#hiring*, which is a popular topic among students.

Figure B.2: The 6 most coherent topics extracted from the MDA community in US Election 2016.

We list the 6 most coherent topics of the MDA community in Figure B.2. We observe several topics that are related to social media networks. For example, Topic 2 has the word *facebook* and Topic 3 has the word *twitter*. There are also names of news outlets in these generated topics, e.g. *@fox* in Topic 3 and *#nbc* in Topic 4. Moreover, we also observe some common topics, such as *nominees* in Topic 1, *#debates* in Topic 4 and *#job* in Topic 6.

Figure B.3 shows the 6 most coherent topics from the BE community. These 6 topics are all related to business. For example, we observe *shop* in Topic 1, *#job* in Topic 2, *businesssupport* in Topic 3, *#careerarc* in Topic 4, *#womeninbusiness* in Topic 5 and *#businessmgmt* in Topic 6. Indeed, the Twitter users in the BE community seem to focusing on business-related discussions during the election.

Finally, we present the 6 most coherent topics in the PLT community in Figure B.4. We observe a topic that is related to *hillary emails* (Topic 1), which is also a controversial topic shown in Section 7.6. Topic 2 is related to *immigrants*. Topic 3 turns to the discussion of *tax*. Topic 6 again mentions the familiar discussion of the *fbi* and its *investigation*.

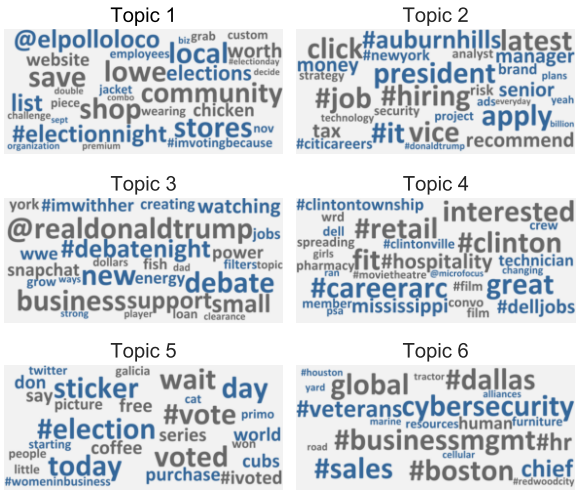

Figure B.3: The 6 most coherent topics extracted from the BE community in US Election 2016.

Figure B.4: The 6 most coherent topics extracted from the PLT community in US Election 2016.

In summary, we observe that all the four communities engaged in many election-related discussions during the election. We also note that some topics were discussed across several communities, such as the topic of *#jobs*. There are also topics that are discussed only in their own communities, such as the tuition fees topic in the ACA community and the topic related to *facebook* in the MDA community. The above analysis provides some insights about the discussed topics among the four communities during the election. A further study can be conducted by social scientists to understand how these four communities responded to the election and the dynamics across communities.

Figure B.5: Topics (7-12) extracted from proClinton using the classical LDA approach.

Figure B.6: Topics (13-18) extracted from proClinton using the classical LDA approach.

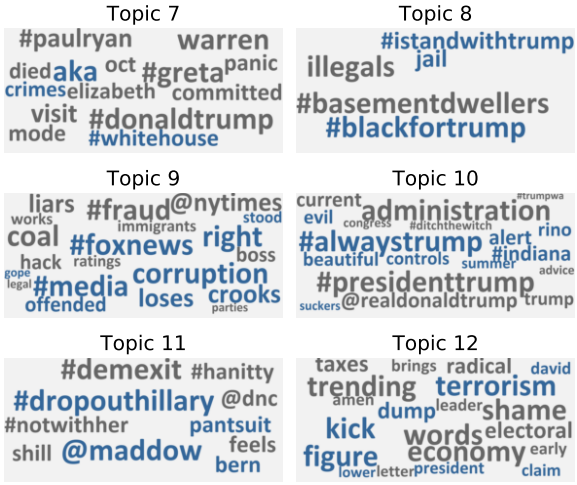

Figure B.7: Topics (7-12) extracted from proTrump using the classical LDA approach.

Figure B.8: Topics (13-18) extracted from proTrump using the classical LDA approach.

Bibliography

Aggarwal, C. C. and Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data*, pages 163–222. sections 2.4.4

Agrawal, R., Rajagopalan, S., Srikant, R., and Xu, Y. (2003). Mining newsgroups using networks arising from social behavior. In *Proceedings of the International Conference on World Wide Web*, pages 529–535. sections 2.5.2

Al Zamal, F., Liu, W., and Ruths, D. (2012). Homophily and latent attribute inference: Inferring latent attributes of Twitter users from neighbors. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 387–390. sections 2.5.2, 3.4.1, 3.4.3.1, 3.1, 3.2, 3.4.3.2, 6.2.2.4

Aletras, N. and Chamberlain, B. P. (2018). Predicting Twitter user socioeconomic attributes with network and language information. *arXiv preprint arXiv:1804.04095*. sections 3.1, 3.2, 3.4.3.2

Allcott, H. and Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–236. sections 7.2.2

AlSumait, L., Barbará, D., and Domeniconi, C. (2008). On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the IEEE International Conference on Data Mining series*, pages 3–12. sections 4.5.2.3

AlSumait, L., Barbará, D., Gentle, J., and Domeniconi, C. (2009). Topic significance ranking of LDA generative models. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pages 67–82. sections 2.2.4, 2.3.2, 2.5.1.2, 3.3.1, 4.1, 4.5.2.3, 5.4.3.2

Arora, S., Ge, R., Halpern, Y., Mimno, D., Moitra, A., Sontag, D., Wu, Y., and Zhu, M. (2013). A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the International Conference on Machine Learning*, pages 280–288. sections 2.3.1

Artstein, R. and Poesio, M. (2008). Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596. sections 4.4.3, 6.2.2.5

Arun, R., Suresh, V., Madhavan, C. V., and Murthy, M. N. (2010). On finding the natural number of topics with Latent Dirichlet Allocation: Some observations. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 391–402. sections 3.3.1

Ashcroft, L. (2014). How scotland voted, and why. `http://lordashcroftpolls.com/2014/09/scotland-voted/`. Accessed October, 2018. sections 6.2.1.2

Asuncion, A., Welling, M., Smyth, P., and Teh, Y. W. (2009). On smoothing and inference for topic models. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 27–34. sections 2.2.3

Bagdouri, M. and Oard, D. W. (2015). Profession-based person search in microblogs: Using seed sets to find journalists. In *Proceedings of the ACM International on Conference on Information and Knowledge Management*, pages 593–602. sections 3.4.2, 3.2

Baker, L. D. and McCallum, A. K. (1998). Distributional clustering of words for text classification. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 96–103. sections 2.2.1

Banko, M. and Brill, E. (2001). Mitigating the paucity-of-data problem: Exploring the effect of training corpus size on classifier performance for natural language processing. In *Proceedings of the International Conference on Human Language Technology Research*, pages 1–5. sections 3.4.1

Bara, J., Weale, A., and Bicquelet, A. (2007). Analysing parliamentary debate with computer assistance. *Swiss Political Science Review*, 13(4):577–605. sections 1.1

Barberá, P. (2015). Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. *Political Analysis*, 23(1):76–91. sections 2.5, 2.5.1.1

Barberá, P. (2016). Less is more? how demographic sample weights can improve public opinion estimates based on Twitter data. Working paper http://pablobarbera.com/static/less-is-more.pdf. sections 2.5.2, 3.1, 3.2

Barberá, P., Jost, J. T., Nagler, J., Tucker, J. A., and Bonneau, R. (2015). Tweeting from left to right: Is online political communication more than an echo chamber? *Psychological science*, 26(10):1531–1542. sections 1.1, 2.5.1.1, 3.4.2

Barber, P. (2015). Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. *Political Analysis*, 23(1):7691. sections 3.2

BBC (2014). Pound falls on fears of scottish independence. `https://www.bbc.co.uk/news/business-29103445`. Accessed October, 2018. sections 5.2.1

Benton, A., Arora, R., and Dredze, M. (2016). Learning multiview embeddings of twitter users. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 14–19. sections 3.4.3.2

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). DBpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165. sections 6.2.2.2

Blake, A. (2017). James comeys fateful decision on hillary clintons emails is slowly coming into focus. `https://www.washingtonpost.com/news/the-fix/wp/2018/01/31/james-comeys-fateful-decision-on-hillary-clintons-emails-is-slowly-coming-into-focus`. Accessed October, 2018. sections 7.6.1, 7.6.2

Blei, D. M. and Jordan, M. I. (2004). Variational methods for the dirichlet process. In *Proceedings of the International Conference on Machine Learning*, pages 1–12. sections 2.2.3

Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the International Conference on Machine Learning*. sections 2.2.1, 2.2.2, 8.2

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *the Journal of machine Learning research*, 3:993–1022. sections 2, 2.2.1, 2.2.4, 2.1, 2.3, 2.5.1.2, 5.4.2

Bowman, K. and Shenton, L. (2004). Estimation: Method of moments. *Encyclopedia of statistical sciences*, 3:2092–2098. sections 5.3.1

Boyd-Graber, J., Hu, Y., Mimno, D., et al. (2017). Applications of topic models. *Foundations and Trends® in Information Retrieval*, 11(2-3):143–296. sections 2.5.1.2

Braun, M. and McAuliffe, J. (2010). Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association*, 105:324–335. sections 2.2.3

Brigadir, I., Greene, D., and Cunningham, P. (2015). Analyzing discourse communities with distributional semantic models. In *Proceedings of the ACM Web Science Conference*, page 27. sections 3.5

Burger, J. D., Henderson, J., Kim, G., and Zarrella, G. (2011). Discriminating gender on Twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1301–1309. sections 3.1, 3.2

Burnap, P., Gibson, R., Sloan, L., Southern, R., and Williams, M. (2016). 140 characters to victory?: Using Twitter to predict the uk 2015 general election. *Electoral Studies*, 41:230–233. sections 1.1, 1.2, 2.5.2

Canini, K., Shi, L., and Griffiths, T. (2009). Online inference of topics with Latent Dirichlet Allocation. In *Proceedings of the Conference Artificial Intelligence and Statistics*, pages 65–72. sections 2.2.1

Cantrell, P. D. (1992). Opinion polling and american democratic culture. *International Journal of Politics, Culture, and Society*, 5(3):405–437. sections 8.2

Cao, J., Xia, T., Li, J., Zhang, Y., and Tang, S. (2009). A density-based method for adaptive LDA model selection. *Neurocomputing*, 72(7-9):1775–1781. sections 3.3.1

Carterette, B., Bennett, P. N., Chickering, D. M., and Dumais, S. T. (2008). Here or there. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 16–27. sections 4.4

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., and Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 288–296. sections 1.1, 1.2, 2.2.4, 2.5.1.2, 3.3.2

Chen, X., Wang, Y., Agichtein, E., and Wang, F. (2015). A comparative study of demographic attribute inference in Twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 590–593. sections 2.5.2, 3.4.1

Cheng, X., Yan, X., Lan, Y., and Guo, J. (2014). Btm: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):2928–2941. sections 3.2.2.2, 4.6

Chris, D. P. et al. (1990). Another stemmer. In *Proceedings of the ACM SIGIR Forum*, volume 24, pages 56–61. sections 2.4.2

Cohen, R. and Ruths, D. (2013). Classifying political orientation on Twitter: It's not easy! In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 91–99. sections 1.2, 2.5.2, 3.4.3.1, 3.1, 3.2, 3.4.3.2

Cong, Y., Chen, B., Liu, H., and Zhou, M. (2017). Deep Latent Dirichlet Allocation with topic-layer-adaptive stochastic gradient riemannian MCMC. In *Proceedings of the International Conference on Machine Learning*, pages 864–873. sections 2.3.1

Conover, M., Ratkiewicz, J., Francisco, M. R., Gonçalves, B., Menczer, F., and Flammini, A. (2011a). Political polarization on Twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 89–96. sections 3.1, 3.2

Conover, M. D., Gonçalves, B., Ratkiewicz, J., Flammini, A., and Menczer, F. (2011b). Predicting the political alignment of Twitter users. In *Proceedings of the IEEE International Conference on Privacy, Security, Risk and Trust and IEEE Conference on Social Computing*, pages 192–199. sections 6.2.2.4

Corasaniti, N. (2016). A look at trumps immigration plan, then and now. `https://www.nytimes.com/interactive/2016/08/31/us/politics/donald-trump-immigration-changes.html`. Accessed October, 2018. sections 7.6.2

Culotta, A., Kumar, N. R., and Cutler, J. (2015). Predicting the demographics of Twitter users from website traffic data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 72–78. sections 3.4.3.1, 3.1

Daiber, J., Jakob, M., Hokamp, C., and Mendes, P. N. (2013). Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the International Conference on Semantic Systems*, pages 121–124. sections 6.2.2.2

Damerau, F. J., Johnson, D. E., and Buskirk Jr, M. C. (2004). Automatic labeling of unlabeled text data. US Patent 6,697,998. sections 3.4.2

De Choudhury, M., Diakopoulos, N., and Naaman, M. (2012). Unfolding the event landscape on Twitter: classification and exploration of user categories. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 241–244. sections 3.4.3.1, 3.1, 3.2, 3.4.3.2

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391. sections 2.2.1

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013). Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 198–206. sections 1.2

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. sections 4.3.2

Diao, Q., Jiang, J., Zhu, F., and Lim, E.-P. (2012). Finding bursty topics from microblogs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 536–544. sections 3.2.2.1

Ding, Y. (2011). Scientific collaboration and endorsement: Network analysis of coauthorship and citation networks. *Journal of informetrics*, 5(1):187–203. sections 2.5.1.1

Enli, G. (2017). Twitter as arena for the authentic outsider: exploring the social media campaigns of trump and clinton in the 2016 us presidential election. *European Journal of Communication*, 32(1):50–61. sections 1.1, 1.2, 2.5.2, 8.3

Euractiv and Reuters (2014). Uk says independent scotland would lose the pound. `https://www.euractiv.com/section/languages-culture/news/uk-says-independent-scotland-would-lose-the-pound/`. Accessed October, 2018. sections 5.2.1

Fagerland, M. W., Lydersen, S., and Laake, P. (2013). The mcnemar test for binary matched-pairs data: mid-p and asymptotic are better than exact conditional. *BMC medical research methodology*, 13(1):91. sections 6.4.4

Fang, A. (2017). Examining information on social media: Topic modelling, trend prediction and community classification. In *Proceedings of the Doctoral Consortium in the International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 1377. sections 1.5

Fang, A., Habel, P., Ounis, I., and MacDonald, C. (2018a). Votes on Twitter: assessing candidate preferences and topics of discussion during the 2016 U.S. presidential election. *SAGE Open,* `DOI:10.1177/2158244018791653`. sections 1.5

Fang, A., Macdonald, C., Ounis, I., and Habel, P. (2016a). Examining the coherence of the top ranked tweet topics. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 825–828. sections 1.5

Fang, A., Macdonald, C., Ounis, I., and Habel, P. (2016b). Topics in tweets: A user study of topic coherence metrics for Twitter data. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 492–504, Padua, Italy. sections 1.5

Fang, A., Macdonald, C., Ounis, I., and Habel, P. (2016c). Using word embedding to evaluate the coherence of topics from Twitter data. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1057–1060. sections 1.5

Fang, A., Macdonald, C., Ounis, I., Habel, P., and Yang, X. (2017). Exploring time-sensitive variational bayesian inference LDA for social media data. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 252–265. sections 1.5

Fang, A., Ounis, I., Habel, P., Macdonald, C., and Limsopatham, N. (2015a). Topic-centric classification of Twitter user's political orientation. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 791–794. sections 1.5

Fang, A., Ounis, I., Habel, P., Macdonald, C., and Limsopatham, N. (2015b). Topic-centric classification of Twitter user's political orientation. In *Proceedings of the Symposium on Future Directions in Information Access*, pages 791–794. sections 1.5

Fang, A., Ounis, I., MacDonald, C., Habel, P., Xiong, X., and Yu, H.-T. (2018b). An effective approach for modelling time features for classifying bursty topics on Twitter. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, pages 1547–1550. sections 1.5, 8.2

Färber, M., Ell, B., Menne, C., and Rettinger, A. (2015). A comparative survey of dbpedia, freebase, opencyc, wikidata, and yago. *Semantic Web Journal*, 1:1–5. sections 6.2.2.2

Fellbaum, C. (1998). *WordNet*. Wiley Online Library. sections 3.3.2

Feltwell, T., Mahoney, J., and Lawson, S. (2015). Aye, have a dream# indyref: use of instagram during the scottish referendum. In *Proceedings of the British Human Computer Interaction Conference*, pages 267–268. sections 1.1

Fisher, M. (2016). The making of sean hannity: How a long island kid learned to channel red-state rage. `https://www.washingtonpost.com/lifestyle/style/the-making-of-sean-hannity-how-a-long-island-kid-learned-to-channel-red-state-rage/2017/10/09/540cfc38-8821-11e7-961d-2f373b3977ee_story.html`. Accessed October, 2018. sections 7.6.1

Fleiss, J. L., Levin, B., and Paik, M. C. (2003). *Statistical methods for rates and proportions*. John Wiley & Sons. sections 6.2.2.5

Gildea, D. and Hofmann, T. (1999). Topic-based language models using em. In *Proceedings of the European Conference on Speech Communication and Technology*, pages 2167–2170. sections 2.3.1

Gilks, W. R., Richardson, S., and Spiegelhalter, D. (1995). *Markov chain Monte Carlo in practice*. Chapman and Hall/CRC. sections 2.2.2

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12). sections 3.4.2

Grčar, M., Cherepnalkoski, D., Mozetič, I., and Novak, P. K. (2017). Stance and influence of Twitter users regarding the brexit referendum. *Computational social networks*, 4(1):6–31. sections 7.6.3, 8.2

Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235. sections 5.4.2

Gruber, A., Weiss, Y., and Rosen-Zvi, M. (2007). Hidden topic markov models. In *Proceedings of the Conference on Artificial Intelligence and Statistics*, pages 163–170. sections 3.2.2.1

Guess, A., Nyhan, B., and Reifler, J. (2018). Selective exposure to misinformation: Evidence from the consumption of fake news during the 2016 u.s. presidential campaign. Working Paper available at `http://www.dartmouth.edu/~nyhan/fake-news-2016.pdf`. Accessed October, 2018. sections 7.2.2

Guolo, A., Varin, C., et al. (2014). Beta regression for time series analysis of bounded data. *The Annals of Applied Statistics*, 8:74–88. sections 5.2.1

Habel, P., Moon, R., and Fang, A. (2018). News and information leadership in the digital age. *Information, Communication & Society*, 21(11):1604–1619. sections 8.2

Habel, P. D. (2012). Following the opinion leaders? the dynamics of influence among media opinion, the public, and politicians. *Political Communication*, 29(3):257–277. sections 1.1

Harding, L. (2016). Top democrat's emails hacked by russia after aide made typo, investigation finds. `https://www.theguardian.com/us-news/2016/dec/14/dnc-hillary-clinton-emails-hacked-russia-aide-typo-investigation-finds`. Accessed October, 2018. sections 7.6.1

Hillman, A. J., Keim, G. D., and Schuler, D. (2004). Corporate political activity: A review and research agenda. *Journal of Management*, 30(6):837–857. sections 1.1

Hillygus, D. S. (2011). The evolution of election polling in the united states. *Public opinion quarterly*, 75(5):962–981. sections 8.2

Hillygus, D. S. and Jackman, S. (2003). Voter decision making in election 2000: Campaign effects, partisan activation, and the clinton legacy. *American Journal of Political Science*, 47(4):583–596. sections 1.1

Hinton, G. E. and Salakhutdinov, R. R. (2009). Replicated softmax: an undirected topic model. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 1607–1614. sections 2.3.1

Hoffman, M., Bach, F. R., and Blei, D. M. (2010). Online learning for Latent Dirichlet Allocation. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 856–864. sections 2.2.2, 2.2.3, 3.5

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 289–296. sections 2.2.1

Hong, L. and Davison, B. D. (2010). Empirical study of topic modeling in Twitter. In *Proceedings of the first Workshop on Social Media Analytics*, pages 80–88. sections 1.1, 1.2, 3.2.1, 3.2.2.2, 5.1

Howard, P. N., Woolley, S., and Calo, R. (2018). Algorithms, bots, and political communication in the us 2016 election: The challenge of automated political communication for election law and administration. *Journal of Information Technology and Politics*, 15(2):81–93. sections 7.2.2

Hussain, J. and Islam, M. A. (2016). Evaluation of graph centrality measures for tweet classification. In *Proceedings of the International Conference on Computing, Electronic and Electrical Engineering*, pages 126–131. sections 3.2

Jacobi, C., van Atteveldt, W., and Welbers, K. (2016). Quantitative analysis of large amounts of journalistic texts using topic modelling. *Digital Journalism*, 4(1):89–106. sections 1.1, 2.5.1.2

Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the Association for Computational Linguistics and Chinese Language Processing*, pages 19–33. sections 3.3.2, 4.1

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*, pages 137–142. sections 2.4.4

Judea, A., Schütze, H., and Brügmann, S. (2014). Unsupervised training set generation for automatic acquisition of technical terminology in patents. In *Proceedings of the COLING International Conference on Computational Linguistics*, pages 290–300. sections 3.4.2

Karami, A., Dahl, A. A., Turner-McGrievy, G., Kharrazi, H., and Shaw, G. (2018). Characterizing diabetes, diet, exercise, and obesity comments on Twitter. *International Journal of Information Management*, 38(1):1–6. sections 2.3.1

Kenter, T. and De Rijke, M. (2015). Short text similarity with word embeddings. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, pages 1411–1420. sections 4.3.2

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751. sections 8.2

Klebanov, B. B., Diermeier, D., and Beigman, E. (2008). Automatic annotation of semantic fields for political science research. *Journal of Information Technology & Politics*, 5(1):95–120. sections 2.5.1.2

Kong, S., Mei, Q., Feng, L., Ye, F., and Zhao, Z. (2014). Predicting bursts and popularity of hashtags in real-time. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 927–930. sections 5.2.1

Kruse, R., Borgelt, C., Klawonn, F., Moewes, C., Steinbrecher, M., and Held, P. (2013). Multi-layer perceptrons. In *Computational Intelligence*, pages 47–81. sections 2.4.4

Kuczma, M. (2009). *An introduction to the theory of functional equations and inequalities: Cauchy's equation and Jensen's inequality*. Springer Science & Business Media. sections 5.2.3.1

Kwon, N., Zhou, L., Hovy, E., and Shulman, S. W. (2007). Identifying and classifying subjective claims. In *Proceedings of the Annual International Conference on Digital Government Research: Bridging Disciplines & Domains*, pages 76–81. sections 2.5.2

Landauer, T. K., Foltz, P. W., and Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, 25(2-3):259–284. sections 3.3.2

Leacock, C. and Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283. sections 3.3.2, 4.1

Lebret, R. and Collobert, R. (2015). N-gram-based low-dimensional representation for document classification. In *Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications*, pages 240–248. sections 4.3.2

Lee, D. D. and Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 556–562. sections 1

Lee, K., Palsetia, D., Narayanan, R., Patwary, M. M. A., Agrawal, A., and Choudhary, A. (2011). Twitter trending topic classification. In *Proceedings of the IEEE International Conference on Data Mining*, pages 251–258. sections 3.2.2.3, 3.2

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the ACM International Conference on Design of Communication by Sigdoc Conference Committee*, pages 24–26. sections 3.3.2, 4.1

Lewis, D. D. (1998). Naive (bayes) at forty: The independence assumption in information retrieval. In *Proceedings of the European Conference on Machine Learning*, pages 4–15. sections 2.4.4

Li, C., Wang, H., Zhang, Z., Sun, A., and Ma, Z. (2016). Topic modeling for short texts with auxiliary word embeddings. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 165–174. sections 3.2.2.2, 3.2.2.3

Li, W. and McCallum, A. (2006). Pachinko allocation: DAG-structured mixture models of topic correlations. In *Proceedings of the International Conference on Machine Learning*, pages 577–584. sections 2.2.1

Liu, S., Wu, Y., Wei, E., Liu, M., and Liu, Y. (2013). Storyflow: Tracking the evolution of stories. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2436–2445. sections 8.2

Liu, T., Chen, Z., Zhang, B., Ma, W.-y., and Wu, G. (2004). Improving text classification using local latent semantic indexing. In *Proceedings of the IEEE International Conference on Data Mining*, pages 162–169. sections 2.2.1

Liu, Y., Niculescu-Mizil, A., and Gryc, W. (2009). Topic-link lda: joint models of topic and author community. In *Proceedings of the Annual International Conference on Machine Learning*, pages 665–672. sections 2.3.2

Llewellyn, C. and Cram, L. (2016). Brexit? analyzing opinion on the uk-eu referendum within Twitter. In *Proceedings of the International Conference on Weblogs and Social Media*, pages 760–761. sections 1.1

LoBianco, T. (2015). Trump 'postpones' israel trip after netanyahu criticism. `https://edition.cnn.com/2015/12/10/politics/donald-trump-postpones-israel-trip/index.html`. Accessed October, 2018. sections 7.6.2

Lovins, J. B. (1968). Development of a stemming algorithm. *Mech. Translat. & Comp. Linguistics*, 11(1-2):22–31. sections 2.4.2

Lu, R. and Yang, Q. (2012). Trend analysis of news topics on Twitter. *International Journal of Machine Learning and Computing*, 2(3):327–332. sections 3.2.2.3

Lucas, C., Nielsen, R. A., Roberts, M. E., Stewart, B. M., Storer, A., and Tingley, D. (2015). Computer-assisted text analysis for comparative politics. *Political Analysis*, 23(2):254–277. sections 2.5.1.1, 2.5.1.2

Ma, Z., Sun, A., and Cong, G. (2013). On predicting the popularity of newly emerging hashtags in Twitter. *Journal of the American Society for Information Science and Technology*, 64(7):1399–1410. sections 5.2.1

Macdowall, C. (2014). How Twitter is being used in the scottish independence referendum debate. `https://phys.org/news/2014-01-twitter-scottish-independence-referendum-debate.html`. Accessed October, 2018. sections 6.2.1

Mackie, S., McCreadie, R., Macdonald, C., and Ounis, I. (2014). On choosing an effective automatic evaluation metric for microblog summarisation. In *Proceedings of the Information Interaction in Context Symposium*, pages 115–124. sections 4.4

Madsen, K. (1973). A root-finding algorithm based on newton's method. *BIT Numerical Mathematics*, 13(1):71–75. sections 8

Manning, C. D., Raghavan, P., and Schütze, H. (2008a). *Introduction to information retrieval*, volume 39. Cambridge University Press. sections 2.4.2, 4.6.1

Manning, C. D., Raghavan, P., Schütze, H., et al. (2008b). Introduction to information retrieval. 1(1). sections 2.2.1

Mascaro, C. M., Novak, A. N., and Goggins, S. P. (2012). The daily brew: The structural evolution of the coffee party on facebook during the 2010 united states midterm election season. *Journal of Information Technology & Politics*, 9(3):234–253. sections 8.2

McCallum, A., Corrada-Emmanuel, A., and Wang, X. (2005). Topic and role discovery in social networks. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, pages 786–791. sections 2.5.1.1

McCallum, A., Nigam, K., et al. (1998). A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI workshop on Learning for Text Categorization*, volume 752, pages 41–48. sections 2.4.4

McCreadie, R., Macdonald, C., and Ounis, I. (2010). Crowdsourcing a news query classification dataset. In *Proceedings of the ACM SIGIR Workshop on Crowdsourcing for Search Evaluation*, pages 31–38. sections 3.4.1

McDonald, G., Macdonald, C., Ounis, I., and Gollins, T. (2014). Towards a classifier for digital sensitivity review. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 500–506. sections 3.4.1

Mehrabian, L. (1998). Effects of poll reports on voter preferences. *Journal of Applied Social Psychology*, 28(23):2119–2130. sections 1.1

Mehrotra, R., Sanner, S., Buntine, W., and Xie, L. (2013). Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. sections 3.2, 3.2.2.2

Mei, Q., Shen, X., and Zhai, C. (2007). Automatic labeling of multinomial topic models. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 490–499. sections 2.2.1, 3.3.1

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011). DBpedia spotlight: Shedding light on the web of documents. In *Proceedings of the International Conference on Semantic Systems*, pages 1–8. sections 6.2.2.2

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781. sections 4.3.2

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 3111–3119. sections 3.5, 4.3.2

Mikolov, T., Yih, W.-t., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 746–751. sections 4.3.2

Miller, G. (1998). *WordNet: An electronic lexical database*. MIT press. sections 4.2

Mimno, D., Wallach, H. M., Naradowsky, J., Smith, D. A., and McCallum, A. (2009). Polylingual topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 880–889. sections 2.3.1

Minka, T. and Lafferty, J. (2002). Expectation-propagation for the generative aspect model. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 352–359. sections 2.3.1

Mislove, A., Lehmann, S., Ahn, Y.-Y., Onnela, J.-P., and Rosenqueist, J. N. (2011). Understanding the demographics of Twitter users. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 554–557. sections 2.5.2

Mladenic, D. and Grobelnik, M. (1999). Feature selection for unbalanced class distribution and naive bayes. In *Proceedings of the International Conference on Machine Learning*, volume 99, pages 258–267. sections 2.4.3, 6.4.4

Moon, T. K. (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60. sections 2.2.3

Morgan-Lopez, A. A., Kim, A. E., Chew, R. F., and Ruddle, P. (2017). Predicting age groups of Twitter users based on language and metadata features. *PloS one*, 12(8):e0183537. sections 3.1, 3.2, 3.4.3.2

Morstatter, F., Pfeffer, J., Liu, H., and Carley, K. M. (2013). Is the sample good enough? comparing data from Twitter's streaming api with Twitter's firehose. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 400–408. sections 23, 7.2

Newman, D., Karimi, S., and Cavedon, L. (2009). External evaluation of topic models. In *Proceedings of the Australasian Document Computing Symposium*, pages 11–18. sections 3.3.2, 4.2

Newman, D., Lau, J. H., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 100–108. sections 2.2.4, 2.1, 2.5.1.2, 3.3.2, 4.2, 4.1, 4.5.2.3, 16

Newman, D. J. and Block, S. (2006). Probabilistic topic decomposition of an eighteenth-century american newspaper. *Journal of the Association for Information Science and Technology*, 57(6):753–767. sections 2.2.1

Nguyen, D. Q., Billingsley, R., Du, L., and Johnson, M. (2015). Improving topic models with latent feature word representations. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 299–313. sections 3.2.2.3

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 79–86. sections 2.4

Papadimitriou, C. H., Raghavan, P., Tamaki, H., and Vempala, S. (2000). Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235. sections 3.4.3.1

Parmelee, J. H. and Bichard, S. L. (2011). *Politics and the Twitter Revolution: How Tweets Influence the Relationship Between Political Leaders and the Public*. Lexington Books. sections 6.4.5

Patterson, S. and Teh, Y. W. (2013). Stochastic gradient riemannian langevin dynamics on the probability simplex. In *Proceedings of the conference in Neural Information Processing Systems*, pages 3102–3110. sections 2.3.1

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). Wordnet:: Similarity: measuring the relatedness of concepts. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 38–41. sections 4.5.2.3

Pennacchiotti, M. and Popescu, A.-M. (2011). A machine learning approach to Twitter user classification. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 281–288. sections 2.4, 2.5.2, 3.4.3.1, 3.1, 3.2, 3.4.3.2

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543. sections 4.3.2, 4.5.2.3

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 2227–2237. sections 4.3.2

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137. sections 2.4.2

Preoţiuc-Pietro, D., Lampos, V., and Aletras, N. (2015). An analysis of the user occupational class through Twitter content. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, pages 1754–1764. sections 3.4.3.2

Quan, X., Kit, C., Ge, Y., and Pan, S. J. (2015). Short and sparse text topic modeling via self-aggregation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2270–2276. sections 3.2.2.2

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., and Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science*, 54(1):209–228. sections 2.5.1.2

Ramabhadran, B., Siohan, O., and Sethy, A. (2007). The ibm 2007 speech transcription system for european parliamentary speeches. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition & Understanding*, pages 472–477. sections 2.5.1.2

Ramage, D., Hall, D., Nallapati, R., and Manning, C. D. (2009). Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 248–256. sections 2.3.2, 3.2.2.3

Rao, D., Yarowsky, D., Shreevats, A., and Gupta, M. (2010). Classifying latent user attributes in Twitter. In *Proceedings of the International workshop on Search and Mining User-generated Contents*, pages 37–44. sections 2.5.2, 3.4.3.1, 3.1, 3.2

Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, pages 43–48. sections 3.4.2

Recchia, G. and Jones, M. N. (2009). More data trumps smarter algorithms: Comparing pointwise mutual information with latent semantic analysis. *Behavior research methods*, 41(3):647–656. sections 3.3.2

Rennie, J. D., Shih, L., Teevan, J., and Karger, D. R. (2003). Tackling the poor assumptions of naive bayes text classifiers. In *Proceedings of the International Conference on Machine Learning*, pages 616–623. sections 2.4.4

Roberts, M. E., Stewart, B. M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S. K., Albertson, B., and Rand, D. G. (2014). Structural topic models for open-ended survey responses. *American Journal of Political Science*, 58(4):1064–1082. sections 2.5.1.1

Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 487–494. sections 2.2.2, 2.3.1, 2.3.2, 3.2.2.2

Ruiz, M. E. and Srinivasan, P. (1998). Automatic text categorization using neural networks. In *Proceedings of the ASIS SIG/CR Workshop on Classification Research*, pages 59–72. sections 2.4.4

Rus, V., Lintean, M. C., Banjade, R., Niraula, N. B., and Stefanescu, D. (2013). SEMILAR: The semantic similarity toolkit. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 163–168. sections 3.3.2

Rustagi, M., Prasath, R. R., Goswami, S., and Sarkar, S. (2009). Learning age and gender of blogger from stylistic variation. In *Proceedings of the International Conference on Pattern Recognition and Machine Intelligence*, pages 205–212. sections 3.4.3.1, 3.1, 3.2, 3.4.3.2

Ryoo, J. J. H. and Bendle, N. (2017). Understanding the social media strategies of u.s. primary candidates. *Journal of Political Marketing*, 16(3-4):244–266. sections 2.5.1.2

Shi, B., Lam, W., Jameel, S., Schockaert, S., and Lai, K. P. (2017). Jointly learning word embeddings and latent topics. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 375–384. sections 3.2.2.3

Sides, J. (2016). Is the media biased toward clinton or trump? here is some actual hard data. `https://www.washingtonpost.com/news/monkey-cage/wp/2016/09/20/is-the-media-biased-toward-clinton-or-trump-heres-some-actual-hard-data`. Accessed October, 2018. sections 7.6.2

Sokolova, M., Huang, K., Matwin, S., Ramisch, J., Sazonova, V., Black, R., Orwa, C., Ochieng, S., and Sambuli, N. (2016). Topic modelling and event identification from Twitter textual data. *Computing Research Repository*, abs/1608.02519:1–17. sections 2.5.1.2, 5.1, 5.4.1.2

Soroush, V., Roy, D., and Aral, S. (2018). The spread of true and false news online. *Science*, 359(6380):1146–1151. sections 7.2.2

Sridhar, V. K. R. (2015). Unsupervised topic modeling for short texts using distributed representations of words. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 192–200. sections 3.2.2.3, 4.6

Stevens, K., Kegelmeyer, P., Andrzejewski, D., and Buttler, D. (2012). Exploring topic coherence over many models and many topics. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 952–961. sections 4.6.4

Steyvers, M. and Griffiths, T. (2007). Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440. sections 2.3.2, 4.5.2.1

Steyvers, M., Smyth, P., Rosen-Zvi, M., and Griffiths, T. (2004). Probabilistic author-topic models for information discovery. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 306–315. sections 2.3.2, 2.5.1.2

Su, T., Fang, A., McCreadie, R., Craig, M., and Iadh, O. (2018). On refining Twitter lists as ground truth data for multi-community user classification. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 765–772. sections 3.4.2, 3.4.3.1, 3.1, 6.2.2.3, 6.2.2.5

Thomas, M., Pang, B., and Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 327–335. sections 2.5, 2.5.2, 3.4.3.1, 3.1

Timberg, C. (2016). Russian propaganda effort helped spread fake news during election, experts say. `https://www.washingtonpost.com/business/economy/russian-propaganda-effort-helped-spread-fake-news-during-election-experts-say/2016/11/24/793903b6-8a40-4ca9-b712-716af66098fe_story.html`. Accessed October, 2018. sections 7.2.2

Vaccari, C., Valeriani, A., Barberá, P., Bonneau, R., Jost, J. T., Nagler, J., and Tucker, J. (2013). Social media and political communication: a survey of Twitter users during the 2013 italian general election. *Rivista italiana di scienza politica*, 43(3):381–410. sections 1.1, 3.4.3.1, 5.4.1.2

Vicente, M., Batista, F., and Carvalho, J. P. (2019). Gender detection of Twitter users based on multiple information sources. *Interactions Between Computational Intelligence and Mathematics Part 2*, pages 39–54. sections 3.4.3.1, 3.1, 3.2, 3.4.3.2

Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009). Evaluation methods for topic models. In *Proceedings of the International Conference on Machine Learning*, pages 1105–1112. sections 2.3.1

Wang, X. and McCallum, A. (2006). Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 424–433. sections 3.2.2.3, 5.3.1

Wei, X. and Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185. sections 2.2.1

Weng, J., Lim, E.-P., Jiang, J., and He, Q. (2010). Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 261–270. sections 3.2.2.2

Weston, J., Watkins, C., et al. (1999). Support vector machines for multi-class pattern recognition. In *Proceedings of the European Symposium on Artificial Neural Networks*, volume 99, pages 219–224. sections 2.4.3, 6.4.2.1

Wines, M. (2016). All this talk of voter fraud? across u.s., officials found next to none. `https://www.nytimes.com/2016/12/18/us/voter-fraud.html`. Accessed October, 2018. sections 7.6.1

Wood-Doughty, Z., Andrews, N., Marvin, R., and Dredze, M. (2018). Predicting Twitter user demographics from names alone. In *Proceedings of the Second Workshop on Computational Modeling of Peoples Opinions, Personality, and Emotions in Social Media*, pages 105–111. sections 3.1, 3.2, 3.4.3.2

Xia, Y., Tang, N., Hussain, A., and Cambria, E. (2015). Discriminative bi-term topic model for headline-based social news clustering. In *Proceedings of the International Flairs Conference*, pages 311–316. sections 3.2.2.2

Yan, X., Guo, J., Lan, Y., and Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of the International Conference on World Wide Web*, pages 1445–1456. sections 3.2.1, 3.2.2.2, 4.6, 4.6.1

Yang, J. and Leskovec, J. (2011). Patterns of temporal variation in online media. In *Proceedings of the ACM International Conference on Web search and data mining*, pages 177–186. sections 5.2.1

Yang, X., Macdonald, C., and Ounis, I. (2018). Using word embeddings in Twitter election classification. *Information Retrieval Journal*, 21(2-3):183–207. sections 8.3

Yi, X. and Allan, J. (2008). Evaluating topic models for information retrieval. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 1431–1432. sections 2.3

Yilmaz, K. E. and Abul, O. (2018). Inferring political alignments of Twitter users. In *Proceedings of the International Symposium on Networks, Computers and Communications*, pages 1–6. sections 3.4.3.1, 3.1, 3.2, 3.4.3.2

Zhao, W.-X., Jiang, J., He, J., Song, Y., Achananuparp, P., Lim, E.-P., and Li, X. (2011a). Topical keyphrase extraction from Twitter. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 379–388. sections 3.2.2.1

Zhao, W.-X., Jiang, J., Weng, J., He, J., Lim, E.-P., Yan, H., and Li, X. (2011b). Comparing Twitter and traditional media using topic models. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 338–349. sections 2.2.1, 2.2.2, 2.2.4, 2.5.1.2, 3.2, 3.2.1, 3.2.2.1, 3, 3.2.2.2, 4.5.2.1, 5.1, 5.3.2

Zou, W. Y., Socher, R., Cer, D., and Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398. sections 4.3.2

Zubiaga, A., Wang, B., Liakata, M., and Procter, R. (2017). Stance classification of social media users in independence movements. *CoRR*, abs/1702.08388. sections 3.4.3.1, 3.1, 3.4.3.2

Zubir, W. M. A. M., Aziz, I. A., Jaafar, J., and Hasan, M. H. (2017). Inference algorithms in Latent Dirichlet Allocation for semantic classification. In *Proceedings of the Computational Methods in Systems and Software*, pages 173–184. sections 2.3.1