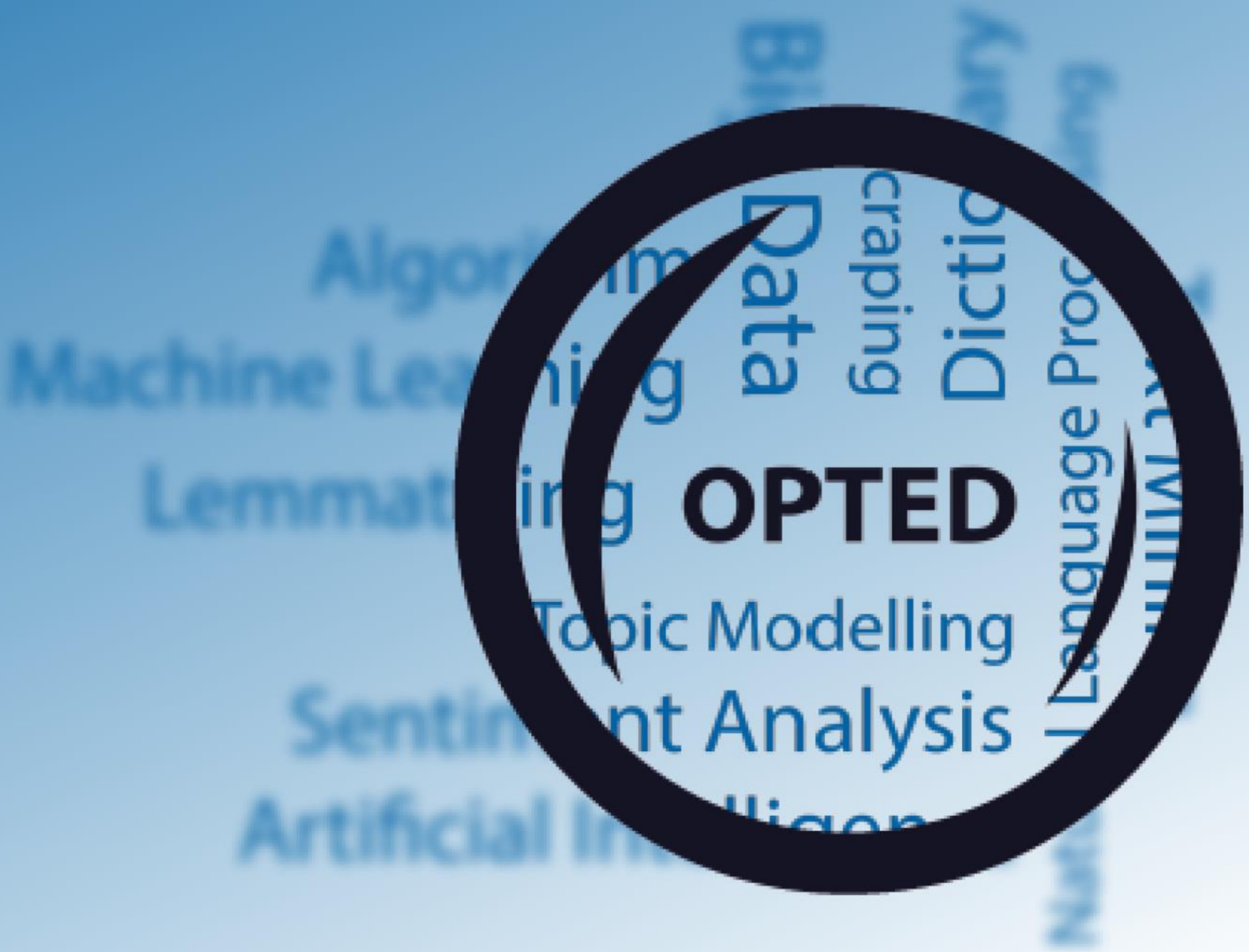
OPTED

Challenges and opportunities for the comparative study of political text types: Testing computer-based topic classification approaches across political party texts

Christoph Ivanusch, Tobias Burst & Lisa Zehnter



Disclaimer

This project has received funding from the European Union’s Horizon 2020 Research & Innovation Action under Grant Agreement no. 951832. The document reflects only the authors’ views. The European Union is not liable for any use that may be made of the information contained herein.

Dissemination level

Public

Type

Report

OPTED
Observatory for Political Texts in European Democracies:
A European research infrastructure

Challenges and opportunities for the comparative study of political text types: Testing computer-based topic classification approaches across political party texts

Deliverable 4.5

Authors: Christoph Ivanusch[1], Tobias Burst[1] & Lisa Zehnter[1]

[1] WZB Berlin Social Science Center

Due date: September 2022

1. Introduction

Text analysis has a long tradition in the social sciences. Especially the communication of political actors and investigations of the dynamics of party competition have attracted significant attention over the past decades. Researchers thereby focus on various types of texts, such as manifestos (e.g. Green-Pedersen, 2007; Guinaudeau and Persico, 2014; Robertson 1976), press releases (e.g. Dalmus et al., 2017; Hopmann et al., 2012), parliamentary speeches (e.g. Debus and Tosun, 2021; Quinn et al., 2010; Rudkowsky et al., 2018), campaign advertising (e.g. Banda, 2015; Damore, 2005) and statements (Sigelman and Buell, 2004) or social media (e.g. Barberá et al., 2019; Gilardi, Gessler, et al., 2022). Although political actors use different text types more or less simultaneously, most existing research studies only one type of text in isolation. Comparative research, hence crossing different text domains, is rare and limited to specific (short) time periods such as election campaigns (Elmelund-Præstekær, 2011; Green and Hobolt, 2008; Norris 1999; Tresch, Lefevere and Walgrave, 2018).

The computational toolbox nowadays contains a wealth of tools to potentially enable such comparative analysis of political communication. Computer-based text analysis (text-as-data) enables researchers to process and analyse large-scale data sets with comparable small costs and time effort. If researchers can apply such methods to study multiple text types, they have powerful tools for analysis at their disposal. This would open up several new avenues for research. For example, researchers would be able to compare communication strategies of political actors in different communication channels or track the evolution of policy preferences at various stages of the political process (i.e. in manifestos, speeches, social media, coalition agreements, legislative texts).

However, combining different text types in a single analysis is a challenging task. The texts produced by political actors are very different in nature and vary on a number of dimensions. Texts can differ with regard to format (e.g. length), style (e.g. complexity of language, symbols/emojis), conventions (e.g. inclusion of quotes) and purpose as texts are produced for different audiences, venues and goals. This variation can influence the applicability of computer-based approaches and hence make analyses across text domains more difficult. Identifying suitable text analysis tools for the comparative study of text types is therefore a challenging task.

A growing amount of literature focuses on the application of computer-based text analysis to study political processes. Existing studies offer valuable, but also narrow insights on the applicability of various computer-based text analysis methods to different types of text. Firstly, researchers are confronted with a variety of text-as-data approaches and specific use cases, but systematic comparisons of the different approaches as well as their performance levels are rare (e.g. Maier et al., 2022). Secondly, applications across text types are only a fairly recent phenomenon and often limited to one specific text-as-data tool (e.g. Burscher et al., 2015; Osnabrügge et. al, 2021). Hence, it remains largely unclear how and to what extent different computer-based approaches can be applied to various types of political communication. What opportunities and limitations result from the application of the individual approaches? Which approaches are suitable for applications across different types of text? Bridging this gap and enhancing our knowledge about the applicability of text-as-data tools across text types would unlock big potential for future research in political science and communication studies.

This deliverable (D4.5) discusses the applicability of computer-based text analysis methods to the comparative study of different text types. First, we provide an overview of existing and widely-used computer-based text analysis methods in the social sciences. This serves as baseline information to later discuss and assess their applicability across text types. We briefly introduce computer-based topic classification, scaling and sentiment analysis approaches.

Then, we discuss the challenges arising from the (comparative) study of different text types. We focus on differences between text types and their implications for the application of computer-based text analysis methods.

Next, we present a case study that illustrates and tests the applicability of text-as-data tools across text types in a practical application. We evaluate and discuss how computer-based topic classification approaches perform on different types of political party texts.[1] We apply unsupervised (LDA), semi-supervised (Newsmap) and supervised (Naïve Bayes, BERT) approaches to manifestos, press releases, parliamentary speeches and tweets from political parties in Austria, Germany and Switzerland (01.01.2019-26.09.2021). The approaches are used to perform a topic classification task based on an adapted codebook from the Manifesto Project (MARPOR). We then systematically examine the main advantages and limitations of each computer-based topic classification approach as well as their applicability for studying various types of political texts.

Finally, we conclude and reflect on the broader implications of our findings for future research and discuss the potential role of OPTED and its possible services when it comes to enabling (comparative) studies of various different types of political text.

2. Review of text analysis methods

Computer-based text analysis is a thriving area of research in political science and communication studies (e.g. Osnabrügge et al., 2021; Rudkowsky et al., 2018; Welbers et al., 2017; Widmann & Wich, 2022). There are several methods and tools available for different tasks, each offering advantages but also coming with limitations. A basic understanding of established text-as-data methods is key to identifying how different text types and text characteristics affect their application. Since it is not the main focus of this deliverable and it is also (almost) impossible to cover the entire universe of computer-based text analysis, we only briefly present methods for three main tasks: topic classification, scaling and sentiment analysis. All three tasks are widely-used in the social sciences and highly relevant to answer central questions in relation to research on politics and democracy.

[1] Replication materials for the case study are available from the authors.

2.1 Topic classification

Which topics are discussed among political actors and the public – the political agenda – is of great interest for the social sciences. Researchers can apply classification techniques to organize texts into topic categories and learn about the political agenda. Topic classification can thereby be used both for a priori known categories and for exploring unknown categories (Grimmer and Stewart, 2013). While unsupervised learning models are especially suitable to identify unknown categories, dictionaries, semi-supervised (i.e. seeded models) and fully supervised tools can be applied to detect known categories in texts (Grimmer and Stewart, 2013; Watanabe and Zhou, 2022).

Unsupervised classification models (i.e. topic models) such as LDA (Blei et al., 2003) or STM (Roberts et al., 2016) identify a specific number of topics (*k*) as clusters of words based on co-occurrences (Benoit, 2020; Watanabe and Zhou, 2022). The number of topics is thereby a priori provided by the user. Unsupervised topic models have their clear strength in exploration because they are able to learn underlying features of text and identify topics based on that (Grimmer and Stewart, 2013). However, sometimes it can be difficult to interpret the topic model outputs and align them with an existing theoretical framework (Watanabe and Zhou, 2022).

Dictionaries are sets of keywords that are associated with specific predefined (topic) categories. Dictionary methods identify the rate at which these keywords appear in a text and this way allow document classification. Dictionaries are intuitive and widely-used, but they also show several limitations. Firstly, dictionaries are strongly context-dependent. Therefore, they can produce false results if not properly validated or need to be newly created for specific tasks, which is cost- and resource-intensive. Secondly, dictionary analysis usually obtains simple frequency counts of categories. This does not offer a theoretically grounded threshold for measuring the reliability of classification results based on probabilities (Watanabe and Zhou, 2022).

Semi-supervised seeded models use a so-called seed-word dictionary as input (e.g. Watanabe, 2018). Seed-word dictionaries define categories (e.g. topics) and provide relevant terms for each of them. Semi-supervised models use this seed-word dictionary to assign topic labels to the documents. Firstly, the models search for seed-words from the dictionary in all documents and assign labels (dictionary categories) to the documents. This allows the model to estimate the association between the labels and textual features (Watanabe and Zhou, 2022). Based on that, semi-supervised models assign the labels specified in the dictionary to all documents. Creating such a dictionary is, however, not easy and requires domain-specific knowledge. Seed-words need to define the categories and have little ambiguity. The main goal is to include seed-words that match intended documents (*true seed words*) and exclude those that match unintended documents (*false seed words*) (Watanabe and Zhou, 2022). Otherwise the classifier's performance might be significantly deteriorated and deliver wrong results.

In supervised classification, users have to provide a labelled data set to train the models for their task. During the training procedure, supervised models learn to encode the relationships between text features and categories (i.e. topics) via machine-learning. To put it simple, supervised models learn from the training data why documents are classified into specific categories. Hence, they learn about the rules for classification. Based on that, the models are capable of transferring this “knowledge” to previously unseen unlabelled text and automatically classify it into the respective categories (Osnabrügge et al., 2021; Welbers et al., 2017). Users, however, do not only need to provide labelled training data, but also have to decide on the size and features of this training data set. Users might have to split the training data into (sub-)samples to account for imbalanced training data for example. Furthermore, training requires extensive monitoring and validation to find the best-fitting model. Hence, training of supervised models requires significant input in the form of labelled data and intense supervision by the user.

2.2 Scaling

Scaling methods allow to locate (political) actors in ideological space through the analysis of texts (e.g. speeches). This enables researchers to test some of the most important theories of politics such as spatial theories of politics (e.g. Downs, 1957). Grimmer and Stewart (2013) differentiate between unsupervised and supervised scaling techniques.

Unsupervised scaling methods, such as *Wordfish* (Slapin and Proksch, 2008), automatically estimate the spatial location of actors in a low-dimensional political space. By discovering words that distinguish locations on a political spectrum, unsupervised scaling models estimate actor positions in this spectrum. Such models can deliver reliable estimates at small costs. However, the lack of supervision makes it difficult to guarantee that the model really measures the ideological locations of political actors and no other dimension apparent in the texts (Grimmer and Stewart, 2013).

Supervised scaling models, such as *Wordscores* (Laver, Benoit and Garry 2003), make use of reference (training) texts to learn about the specific ideological space. The reference texts represent certain positions in the ideological space (e.g. the left and right poles of the spectrum). The model then uses these reference texts to generate scores for each word. These scores measure the word’s position on the political spectrum (e.g. left-right). Based on that, the remaining (non-reference or “virgin”) texts are scaled. This procedure can be used to scale texts on multiple dimension, but the results are also highly dependent on the provided reference texts.

2.3 Sentiment analysis

Sentiment analysis measures the tonality of communication (e.g. negativity). This way, it enables the study of media tone (Van Atteveldt et al., 2008), public opinion (e.g. González-Bailón & Paltoglou, 2015) or negative campaigning (Haselmayer & Jenny, 2017) for example. Sentiment analysis has gained significant traction in the social sciences since the mid-2000s (Haselmayer, 2019) with several studies using computer-based sentiment analysis techniques. Here, two main approaches can be differentiated: sentiment dictionaries and supervised sentiment analysis.

Sentiment analysis with dictionaries uses both existing off-the-shelf dictionaries (e.g. Young & Soroka, 2012; Valentim & Widmann, 2021) and customised (context-sensitive) dictionaries (e.g. Haselmayer & Jenny, 2017). Creating such dictionaries can deliver good results, but their domain-specificity and the high costs of developing custom dictionaries are potential disadvantages.

Supervised sentiment analysis makes use of labelled data to train different classification algorithms (e.g. Petkevic & Nai, 2022; Rudkowsky et al., 2018; Van Atteveldt et al., 2008). While some older studies rely on a bag-of-words approach for representing text (e.g. Van Atteveldt et al., 2008), more recent applications take grammatical structures and context into account by using semantic models such as word-embeddings for example (e.g., Rudkowsky et al., 2018; Widmann & Wich, 2022)

3. Challenges when studying different types of text

Political actors produce a broad range of texts, such as manifestos, press releases or social media. The different text types thereby vary on a number of dimensions and this variation has several potential effects on the applicability of computer-based text analysis tools.

Firstly, texts from political organizations differ with regard to their format. While manifestos are extensive documents spanning multiple (often hundreds of) pages, tweets are restricted to a maximum of 280 characters for example. Therefore, users need to decide on the unit of analysis. Short texts (e.g. tweets) often have a focus on one particular aspect or topic and may therefore allow document-level analysis; longer texts (e.g. manifestos, parliamentary speeches) regularly touch on several different topics. Thus, it might be necessary to separate texts into individual paragraphs or sentences to enable a meaningful analysis and capture potential different facets of a text. As Baden et al. (2022) argue, social scientists are, for example, often less interested in the overall sentiment of a document as a whole, but rather with regard to a specific topic or country. In such cases, it may be necessary to split longer texts, such as manifestos, into smaller units (e.g. paragraphs or sentences) for a more in-depth analysis. Additionally, it might be useful to take account of contextual information. This is for instance the case in sentence-level classification. Often individual sentences do not have strong indicators that allow reliable classification. Here, surrounding sentences can offer valuable information for classifying the specific sentence (e.g. Watanabe and Zhou, 2022).

Secondly, texts vary regarding their style. Depending on the text, authors might use a different vocabulary (i.e. domain-specific words) and different grammatical structures or include URL links or symbols (e.g. emojis). Furthermore, some text types tend to contain more grammatical or spelling errors than others (e.g. social media). As briefly discussed above in the section on text analysis methods, some tools are highly domain- and context-dependent (e.g. dictionaries, supervised scaling models). Hence, tools that work well for one type of text may not travel well to other types of text. This might be less pronounced for text types with similar style as the usage of specific words does not differ substantially. Both manifestos and press releases, for example, are written communication, often offer dense policy-related information and therefore use a similar vocabulary. Parliamentary speeches as spoken communication and social media as less formal communication on the other hand differ significantly in style and vocabulary from manifestos, for example. Applying the same text-as-data approaches without adjustments across such different text types can result in varying performance and potentially be problematic. Furthermore, users need to decide whether elements such as URL links, symbols or emojis are relevant during the analysis or should be discarded (e.g. emojis can be highly relevant in the case of sentiment analysis, but less so in topic classification). Depending on this decision, different steps during text cleaning and pre-processing are necessary. Including or excluding different elements (e.g. URL links, emojis) can potentially influence computer-based models and the resulting output. Thus, users need to be careful and have to be aware of different text styles and their potential implications for the application of computer-based text analysis tools.

Thirdly, text types follow different conventions. While press releases and social media posts for example regularly include direct quotes, speeches often contain greetings and direct addresses. Such text specifics can severely influence how text-as-data tools need to be applied. For example, researchers may be interested in the topics discussed in parliamentary speeches and want to apply an existing topic category scheme, such as from the Comparative Agendas Project (CAP) or the Manifesto Project (MARPOR). However, in order to not deteriorate the classifier, it may be necessary to add an additional category for greetings and direct addresses. Adding such additional categories during the analysis is thereby easier for some text-as-data approaches (e.g. dictionaries, semi-supervised models) than others such as fully supervised models for example. As discussed earlier, supervised models require (extensive) labelled training data. Here, adding new labels to the coding scheme and to a potentially already existing training data set can be very time- and resource-intensive. Thus, the format of texts and its potential implications need to be considered at different stages of the text analysis workflow (e.g. definition of coding scheme, cleaning and pre-processing). This aspect is especially important when working with several different text types simultaneously.

Fourthly, texts can be produced for different audiences, venues and goals. Thus, text types differ with regard to their purpose and political actors potentially communicate different contents in them. Manifestos for instance offer dense policy-related information. They are negotiated at length inside parties and are viewed as a “uniquely representative and authoritative characterization of party policy at a given point in time” (Budge et al., 1987, p. 18). Press releases and tweets on the other hand do not only contain policy-related content, but also inform journalists and the public about campaign events or organisational matters. Hence, they include a substantial amount of non-policy-related content as well. Meyer, Haselmayer and Wagner (2020) for example exclude nearly 300 out of 1,922 press releases from their study of the Austrian national election campaign in 2013 because the content is not policy-related. Such differences in purpose potentially influence the applicability of computer-based text analysis tools. Similar to the example of greetings and direct addresses in parliamentary speeches discussed above, non-policy-related content (e.g. information about campaign events) might be confused by an algorithm with policy-related content (i.e. political issue, policy position). This can significantly deteriorate the results delivered by a text-as-data tool. Users need to be aware of this and may need to make some adaptations during data preparation and pre-processing or adjust the applied algorithm.

This overview shows that applying computer-based text analysis tools to various types of text is challenging. Users have to be aware of text format, style, conventions and purpose as well as their potential effects on the applicability of specific tools. Hence, domain-specific knowledge and familiarity both with regard to the analysed text types and their content are required. Furthermore, users need sufficient methodological expertise to foresee potential challenges and to implement necessary steps to overcome them.

4. Case study

The following case study illustrates the applicability of text-as-data tools across text types using the example of computer-based topic classification. Topic classification allows researchers to organize texts into topic categories and identify those topics that are important in the political system (e.g. during an election campaign). Therefore, topic classification is an important and widely-used method in the social sciences and various approaches to it exist. We apply unsupervised (LDA), semi-supervised (Newsmap) and supervised (Naïve Bayes, BERT) topic classification tools on different types of political text. The study provides both a qualitative and quantitative comparison of the different approaches and of their applicability to various text types.

The report on this case study is structured as follows. In a first step, we describe the different approaches to topic classification by comparing their analysis workflows in general and in terms of the requirements for studying different types of text. Then, we describe our study design and methodological approach. Finally, we present our findings by discussing the advantages and limitations of the individual approaches as well as how different text types affect their application.

4.1 Topic Classification: Approaches and workflows

Three main approaches to topic classification can be distinguished - unsupervised, semi-supervised and supervised learning. Each of these approaches is characterised by specific principles along which a variety of tools have been developed. While unsupervised tools (e.g. topic models) do require no or only very limited input, semi-supervised and supervised tools rely on some sort of prior information to perform the classification task. However, there are further differences between the approaches. These differences influence the workflow as each approach requires specific inputs and decisions by the user. Figure 1 illustrates and compares the workflows of unsupervised, semi-supervised and supervised approaches to topic classification. We differentiate four main steps in the workflow: data foundation, pre-processing, model building and post-processing. The following paragraphs discuss the core characteristics of each approach with regard to these four steps.[2]

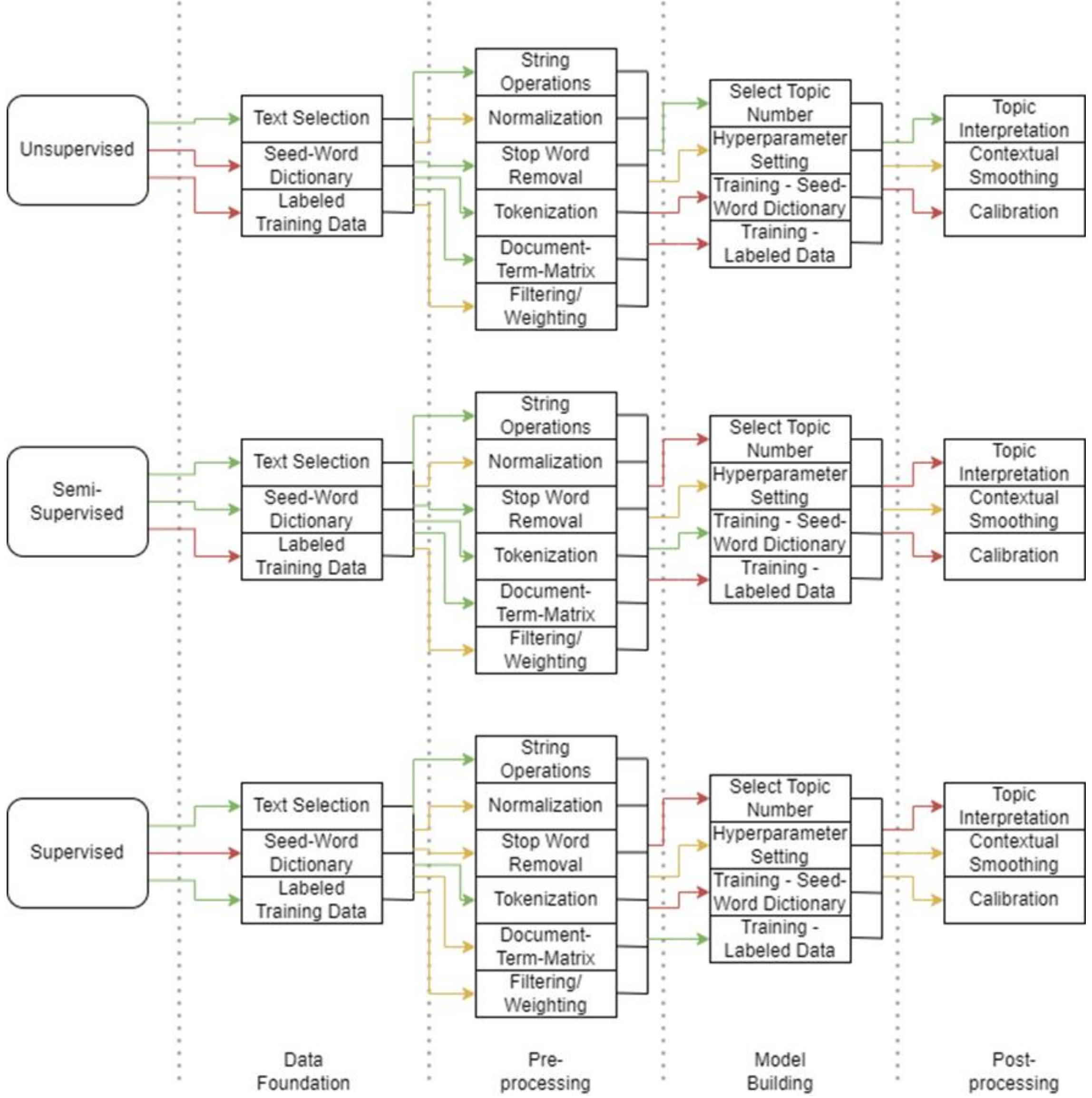
Data foundation

The first step in the workflow for computer-based topic classification relates to the necessary data foundation. Firstly, users need to decide on the documents they want to include as well as the level of analysis (e.g. document-level, sentence-level). This is closely related to the selection of the texts to be analysed and their format (e.g. length). Secondly and crucially, some approaches require additional information or data as

[2] Although we aim to provide a comprehensive overview, we mainly base it on the characteristics of frequently used and/or well-known tools (e.g. topic models, Naive Bayes classifiers) in the social sciences. We do not consider all existing tools for reasons of simplicity and coherence.

input prior to the application. This is also the most important and defining difference between unsupervised, semi-supervised and supervised tools.

Figure 1 COMPARISON OF WORKFLOWS



Note: The arrow colours indicate the relevance of each step in the workflow for the different classification approaches: green - mandatory, yellow - optional/model dependent, red - not applicable.

Unsupervised tools usually require none or only a limited amount of prior information in order to be applied. One of the best-known unsupervised tools in the area of topic classification are topic models such as LDA (Blei et al., 2003) or STM (Roberts et al., 2016). Users only have to specify the number of topics (*k*) that the topic model should recognise in the documents (Benoit, 2020; Watanabe and Zhou, 2022). Hence, unsupervised tools do not require any additionally data, but identifying a suitable number of topics (*k*) can be a complex task, especially when texts differ substantially in format and content.

In contrast to that, supervised tools usually rely on labelled data sets for training. The training data thereby should contain good examples representing the categories (e.g. topics) that the user attempts to predict or measure (Benoit, 2020). Hence, these categories need to be the same as used for the overall classification and each document already needs to be assigned to a topic category. The supervised classifier uses this information to learn the task. Hence, users have to provide labelled training data in order to apply supervised classifiers. As discussed earlier, providing such labelled training data that works for several text types can be challenging, especially if the different types vary on several dimensions (e.g. format, style). Ideally, the labelled texts in the training data set are representative of the texts that are subject to the classification task. This needs to be kept in mind, especially for topic classification across text types.

Semi-supervised tools reside somewhere between these two poles. They require prior information, but usually in a more limited format than supervised tools. Semi-supervised techniques thereby use both labelled and unlabelled documents for training models (Chapelle et al., 2006). While various types of semi-supervised techniques have been developed, two prominent strands can be identified. On the one hand, semi-supervised techniques are used as add-ons to improve the performance of supervised classification (e.g. Banerjee et al., 2007; Phan et al., 2008; Schönhofen, 2009; Zelikovitz and Hirsh, 2000). On the other hand, seeded models are applied to improve the interpretability of results (Watanabe and Zhou, 2022). Users need to provide such models with (small) input data, such as a seed-word dictionary. In the case of topic classification such a seed-word dictionary defines the topic categories and provides relevant seed-words (keywords) for each of the categories. Although developing such a seed-word dictionary requires a substantial amount of knowledge about the specific case, it is also a flexible tool, which allows comparatively quick adaptations to different types of text. For example, users can add or remove categories with relative ease. The resulting seed-word dictionary is then used by a model as input to identify topics in unlabelled documents (e.g. Watanabe and Zhou, 2022). For the purpose of this paper, we only consider such seeded-models as semi-supervised tools.

Pre-processing

The second step concerns the pre-processing of textual data. Users of computer-based text analysis tools have to prepare textual data prior to the actual application and the choices made at this stage can have profound effects on the analysis (Denny & Spirling, 2018). Welbers et al. (2017) identify six important processes: (1) string operations, (2) tokenization, (3) normalization (lowercasing and stemming), (4) removal of stopwords, (5) creation of document-term matrix, (6) filtering and weighting of terms.

Most unsupervised and semi-supervised tools require several of the named pre-processing processes. String operations, tokenization, the removal of stopwords and the creation of a document-term matrix are necessary in nearly all applications. Lowercasing, stemming as well as filtering and weighting of the terms are potential further steps depending on the tool and/or task at hand. Pre-processing for supervised topic classification is even more dependent on the applied tool. While some tools only require tokenization (e.g. BERT), others follow a similar process as most unsupervised and semi-supervised tools (e.g. Naïve Bayes classifiers).

Hence, users need to be aware of pre-processing requirements resulting from each tool and each text type. Including or excluding words or symbols (e.g. emojis) specific to certain text types can, for example, severely influence model performance.

Model building

In a third step, users have to build and/or train the model. As discussed in the section on the necessary data foundation, each approach requires different types of input to build and/or train the models.

In unsupervised topic classification, only limited input and no additional data is required to build the models. In the case of topic models (e.g. LDA) for example, users only need to provide the model with the number of topics (*k*) that it should detect in the documents. The topic models then identify this user-provided number of topics (*k*) as clusters of words based on co-occurrences (Benoit, 2020; Watanabe and Zhou, 2022). Additionally, user can also adapt the setting of specific (hyper)parameters (Maier et al., 2018). Based on that, users can predict the most likely topics for each document.

Building semi-supervised models (i.e. seeded models) in contrast requires a more extensive input in the form of a seed-word dictionary. This additional data defines the categories (e.g. topics) and provides relevant seed-words for each of the categories. Creating such a dictionary is, however, not easy and requires domain-specific knowledge. Seed-words need to define the categories and have little ambiguity. Semi-supervised models use this seed-word dictionary to estimate the association between topic labels (dictionary categories) and textual features and then assign the labels specified in the dictionary to all documents (Watanabe, 2018; Watanabe and Zhou, 2022).

In supervised classification, users have to provide a labelled data set for building the models. This labelled data is necessary to train the models for their task. During the training procedure, supervised models learn to encode the relationships between text features and categories (e.g. topics) via machine-learning. To put it simple, supervised models learn from the training data why documents are classified into specific categories. The models transfer this “knowledge” to previously unseen unlabelled text and automatically classify it into the respective categories (Osnabrügge et al., 2021; Welbers et al., 2017). Users, however, do not only need to provide labelled training data, but also have to decide on the size and features of this training data set. Users might have to split the training data into (sub-)samples to account for imbalanced training data for example. Furthermore, training requires extensive monitoring and validation. This allows users to find the best-fitting model and adapt its parameters to fit the task. Hence, training of supervised models requires significant input and supervision by the user.

Post-processing

The fourth and final step in the workflow concerns different processes after the actual classification. These post-processing processes concern both improving the interpretability as well as the performance of the topic classification. The requirements for such post-processing steps differ between unsupervised, semi-supervised and supervised tools.

Unsupervised tools (i.e. topic models) require extensive post-processing, otherwise their output is difficult to interpret. Unsupervised models provide the most likely topic for each document. However, these topics are solely clusters of words based on co-occurrences (Benoit, 2020; Watanabe and Zhou, 2022). This means that the model does not provide meaningful labels for the different topics. The model output is therefore difficult to interpret and often inconsistent with the theoretical framework, but some techniques to make the output more interpretable exist. On the one hand, users can manually map the topics provided by the model to substantive topic categories. On the other hand, Béchara et al. (2021) propose a new automated transfer topic labelling method. They use a domain-specific codebook as a knowledge base to automatically label topics delivered by an unsupervised topic model. More precisely, this method matches the topics provided by the model with category descriptions from the codebook. This method certainly is a promising avenue to increase the interpretability of topic model outputs. The applicability, however, depends on two main aspects. Firstly, the used codebook and the classified texts need to have substantial overlap with regard to the usage of words in order to successfully label the topics. This may vary across different text types. Secondly, the category descriptions in the codebook need to be of high quality. In contrast, semi-supervised and supervised models do not require such techniques to make the model outputs more interpretable.

Further post-processing can be applied to all three approaches. One technique that is potentially beneficial in the case of all three approaches is contextual smoothing. It is especially relevant for sentence-level classification of longer texts because it allows to take the context (surrounding sentences) into account. Contextual smoothing does not require a modification of the used model itself, but is rather an adaptation of the model output. Nearly all models provide the probability of topics for each document or document part (e.g. sentence). Contextual smoothing of sentence-level classifications draws on these probabilities for each sentence and the respective surrounding sentences. It then uses a kernel smoother to reclassify the sentences into topics with the highest scores. This allows to improve sentence-level classification (Watanabe and Zhou, 2022).

4.2 Study design

For our study, we face several challenges. Firstly, we aim to test different approaches to computer-based topic classification. As introduced above, each approach contains a wealth of tools with specific workflows. Hence, it is imperative to choose tools for our study that are representative of the respective approach. Secondly, we are interested in the applicability of the different approaches to a variety of texts. Hence, we have to select and collect different types of text. Thirdly, we have to identify the best way to apply each computer-based tool. Finally, we need to evaluate the performance of the computer-based approaches. The following sections describe how we address these challenges.

Tool selection

In the case of unsupervised topic classification, we opt for a classic topic model. More precisely, we apply Latent Dirichlet Allocation (LDA), which was introduced by Blei et al. (2003). LDA is a frequently used topic model in political science and communication studies (e.g. Maier et al., 2018). Although unsupervised tools in general may be more suitable for classification tasks with unknown categories (Grimmer and Stewart, 2013), recent applications use topic models with known categories as well (e.g. Béchara etl al., 2021). We therefore also include it in our study. However, as discussed above, interpreting the output of topic models is difficult. We use the automated transfer topic labelling method proposed by Béchara et al. (2021) to assign interpretable labels to the topics delivered by the LDA model.

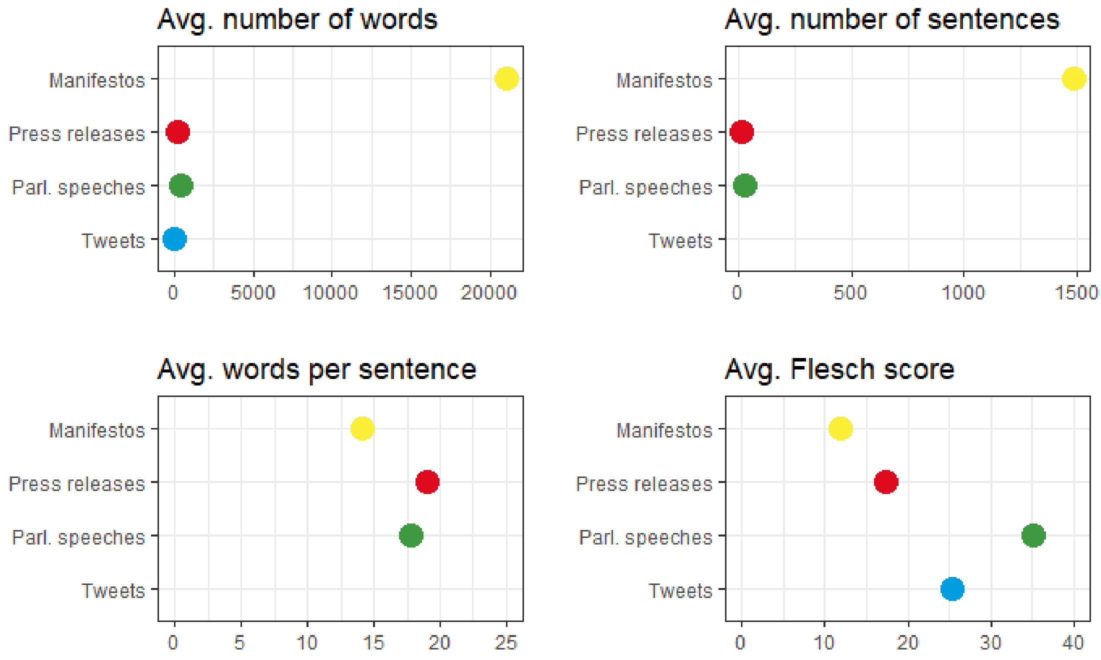
As a semi-supervised tool, we choose Newsmap (Watanabe, 2018). Newsmap is a typical seeded model. In contrast to fully supervised models, no labelled training data set is necessary. Newsmap relies on a seed-word dictionary for classification and was originally developed for geographical classification. However, the use of seed-word dictionaries allows high flexibility and also enables other classification tasks. Watanabe and Zhou (2022) show that Newsmap is also applicable to topic classification and is capable of outperforming other seeded models such as Seeded-LDA for example. Newsmap at first searches the individual documents for keywords in the seed-word dictionary (simple keyword matching) and gives them category labels (e.g. topics). Then, the system aggregates the frequency of words according to the category labels to create contingency tables. Based on that, Newsmap predicts categories (e.g. topics) most strongly associated with documents in the classification stage (Watanabe, 2018).

To evaluate supervised topic classification, we apply two different models: a Naïve Bayes classifier and a BERT model. Naïve Bayes classifiers use a bag-of-words approach and are comparatively simple but reliable algorithms. Therefore, Naïve Bayes classifiers are frequently used when comparing the performance of multiple different models (e.g. Chang and Masterson, 2020; Jerzak et al., 2022). Transformer-based models such as BERT (Devlin et al., 2019) on the other hand are a relatively new development that has revolutionized machine learning in the text-as-data domain in recent years. Transformer-based models have in common that they are elaborately pre-trained on vast amounts of unlabelled text, so their weights are accordingly good at providing a general syntactic and semantic representation of words and can be used directly for subsequent use cases. To use those models for our application, further training (“finetuning”) on labelled text is necessary: The model and its parameters are thereby adjusted to the relevant data and task without training it from scratch. Utilizing this transfer-learning principle, transformer-based models represent the current state-of-the-art in language modelling. Hence, they are potentially also very well-suited for topic classification.

Data

The inclusion of several text types in this study allows an in-depth analysis of the strengths and weaknesses of the different approaches for topic classification and the identification of challenges arising for computer-based text analysis when studying different text types. We therefore analyse manifestos, press releases, parliamentary speeches and tweets from political parties in Austria, Germany and Switzerland. These text types differ on a number of dimensions, such as format, style, conventions and purpose. Figure 2 provides some statistics that illustrate the differences between party manifestos, press releases, parliamentary speeches and tweets. The statistics focus on differences in format (e.g. length) and style (e.g. Flesch score). The Flesch score indicates the readability of a text, i.e. how difficult it is to understand a text.

Figure 2 COMPARISON OF TEXT TYPES USED IN THE CASE STUDY



Notes: The statistics for "Avg. number of sentences" and "Avg. words per sentence" are not available for tweets. This is the case as it is difficult and not reasonable to split tweets into sentences.

We select political texts from Austria, Germany and Switzerland because all three countries are German-speaking. While comparing different types of texts is a significant challenge in itself, a multi-lingual analysis would add another layer of complexity. Furthermore, multi-lingual analyses are challenging and potentially deliver incommensurable results (Baden et al., 2022; Chan et al., 2020; Maier et al., 2022). Therefore, we restrict this study to political parties from German-speaking countries (Austria, Germany and Switzerland).

We rely on webscraping and existing data sets to create the complete text corpus for this study. The observational period ranges from the 1st of January, 2019 until the 26th of September, 2021. We include all parties represented in the respective national parliaments during this time period.

The relevant manifestos (Austria: 2019; Germany: 2021; Switzerland: 2019) are available via the Manifesto Project (MARPOR), which stores manually coded manifestos of more than 1,000 parties from 1945 until today in over 50 countries (Burst et al., 2021). The manifestos are coded based on an extensive codebook covering multiple different political issues. In the case of press releases, we draw on country-specific data sources.[3] All Austrian press releases are available via a webservice from the Austrian Press Agency (APA-OTS, n.d.). German press releases are available via the websites of the parties and their parliamentary party groups. For both Austria and Germany, we use web scraping to collect the press releases. In the case of Switzerland, we use a data set provided by the DigDemLab at the University of Zurich (Gilardi, Baumgartner,

[3] We include press releases published both by the central party offices and the parliamentary party groups.

et al., 2022). With regard to parliamentary speeches, we draw on an updated version of the ParlSpeech V2 data set provided by Rauh and Schwalbach (2020) for Austria and Germany. In the case of Switzerland, we rely on the R package *swissparl* (Zumbach, 2020) to access the webservices of the Swiss parliament and download the speeches. For collecting tweets, we use the Twitter Researcher API. This way we are able to download all tweets published by Austrian, German and Swiss parties during the covered time period.[4] Table 1 provides an overview of the complete corpus used for this study. Overall, the data set consists of more than 147,000 individual documents and over 1.5 million sentences.[5]

Table 1 DOCUMENTS PER TEXT TYPE

Type	Documents	Sentences
Manifestos	23	34,220
Press releases	34,421	462,358
Parl. Speeches	41,497	967,410
Tweets	71,894	71,894
All	147,835	1,535,882

Application

Before the actual application of the different topic classification tools, several decisions have to be made. Firstly, we need to define the classification unit. This decision is mainly influenced by the types of text at hand. Manifestos are for example very extensive documents that cover several different topics. Here, document-level classification is not meaningful. Traditionally, manifesto researchers use sentence-level or quasi-sentence-level classification.[6] Press releases and parliamentary speeches also consist of multiple sentences. They usually have a more specific focus than manifestos, but can still address or touch upon multiple topics and aspects. To ensure comparability of the used text analysis applications across different types of text, we use sentence-level classification in this study. The only exception are tweets as they are too short and unstructured to create a meaningful sentence-level data set. For other research questions or tasks, however, an analysis of the press releases or parliamentary speeches at the document- or paragraph-level might be more useful. Hence, the choice of classification unit is strongly influenced by the specific text types and task at hand.

Secondly, we need to decide on the classification scheme. We use an adapted codebook from the Manifesto Project (Werner et al., 2021). The original codebook consists of 56 main codes and 32 additional sub-category codes. In line with other topic classification research, we merge these fine-grained codes into overarching topic categories. Overall, our codebook consists of 20 topic categories.[7]

We apply unsupervised LDA, semi-supervised Newsmap and supervised Naïve Bayes and BERT models

[4] We include tweets published both by the central party offices and the parliamentary party groups.

[5] We did not split the tweets into individual sentences. Tweets are short and difficult to split into individual sentences because of the text style. Hence, the number of documents is equal to the number of sentences for tweets in this case.

[6] The Manifesto Project (Volkens et al., 2021) classifies manifestos at quasi-sentence-level. The main argument is that natural sentence sometimes contain more than one topic. Coders are allowed to split such natural sentences into quasi-sentences.

[7] Appendix A contains the adapted codebook based on the Manifesto Project coding scheme (Werner et al., 2021).

to the different types of text. For all four tools, we mostly apply standard procedures.[8]

In the case of LDA, we run multiple configurations to identify the best-fitting model. During pre-processing we engage in some limited domain-specific cleaning,[9] split all German compound words (necessary for automated transfer topic labelling during post-processing) and remove all punctuation, numbers, symbols, URLs, separators and stopwords. Then we perform tokenization and stemming. Overall, we run 20 different models per text type. These models result from different combinations. We specify different numbers of topics (k), calculate individual models per country and models for all three countries combined.[10] During post-processing we use automated transfer topic labelling following Béchara et al. (2021) to acquire the topic categories from our codebook. Furthermore, we perform contextual smoothing to denoise the individual sentence-level classification (see: Watanabe and Zhou, 2022).

Newsmap does not require to calculate as many different models. Here, we run two models: one with and one without contextual smoothing. During pre-processing we perform some limited domain-specific cleaning, and remove all punctuation, numbers, symbols, URLs, separators and stopwords. After tokenization, we compound tokens that are present in the seed-word dictionary. We create this dictionary by combining knowledge- and frequency-based seed-words (see Watanabe and Zhou, 2022). Then we apply Newsmap based on the seed-word dictionary and use contextual smoothing for post-processing.

For classification using a Naïve Bayes model, we proceed in four steps. Firstly, we engage in some limited domain-specific cleaning and remove all punctuation, numbers, symbols, URLs, separators and stopwords and then perform tokenization. Secondly, we train the Naïve Bayes model on labelled manifestos from Austria, Germany and Switzerland. We include manifestos published from 2013 onwards[11] , but exclude the most recent ones as they are part of our corpus for classification. Overall, our labelled training data consists of 32,627 (quasi-)sentences. Thirdly, we apply the model trained on manifestos to the respective unlabelled target documents (manifestos, press releases, parliamentary speeches and tweets). Finally, we use contextual smoothing.

BERT can be used as a supervised learning tool in a similar way as Naïve Bayes classifiers. To apply BERT, we use a pre-trained multilingual model obtained via the HuggingFace python library (Wolf et al., 2020). We train ("finetune") this model on the same labelled manifestos as the Naïve Bayes model and then apply the trained BERT model to the unlabelled manifestos, press releases, parliamentary speeches and tweets. In the case of BERT, too, we perform contextual smoothing during post-processing.

Hence, we apply the two supervised classification tools (Naïve Bayes and BERT) in two ways: (1) within-domain to manifestos and (2) cross-domain to press releases, parliamentary speeches and tweets. We select

[8] Appendix B provides detailed descriptions of the applications.

[9] During domain-specific cleaning we remove some text- and context-specific words that potentially deteriorate the classification model. For example, some classification models might confuse sentences including the term "Social Democratic Party" with labour or welfare state issues and therefore classify these sentences wrongly. To prevent such false classification results, we clean the texts during pre-processing. This varies slightly across text types and used classification models.

[10] When applying LDA to tweets, we use slightly different model combinations (see Appendix B and D).

[11] In 2015 the Manifesto Project made some adjustments to the coding scheme. Therefore, we only include manifestos which have been coded after 2015 in our training data set. This also includes the 2013 Austrian manifestos.

this procedure for two main reasons. Firstly, training supervised tools requires extensive high-quality labelled training data. Creating such data is costly, time-consuming and difficult. Therefore, in many cases it is not scalable to develop separate labelled training data for multiple types of tasks or texts. However, large-scale projects such as the Manifesto Project (Volkens et al., 2021) already provide high-quality labelled data sets. These can be used to train supervised models and apply them both within- and cross-domain. Secondly, researchers are increasingly interested into the applicability of cross-domain learning for studying political texts (e.g. Burscher et al., 2015; Osnabrügge et al., 2021). Therefore, we train both supervised models (Naïve Bayes and BERT) on coded manifestos from the Manifesto Project (Volkens et al., 2021) and apply them within- and cross-domain.

Evaluating approaches

To provide a comprehensive evaluation of the different topic classification approaches and their applicability to various text types, we engage in both a qualitative and quantitative comparison. To measure the performance of these computer-based text analysis models on the different text types, we use human coding as a "gold standard". We check the computer-based topic classifications against manual annotation and measure the agreement. This procedure allows us to evaluate the performance levels of each computer-based approach. Such validation practices, which compare computer-based models with human coding, are widely-used in the field of computer-based text analysis (Song et al., 2020). While the "gold standard" for manifestos is already available via the Manifesto Project (Volkens et al., 2021), we rely on trained student coders from the Manifesto Project to create one for the press releases, parliamentary speeches and tweets as well. Overall, our "gold standard" consists of 21 manifestos (29,179 sentences), 346 press releases (4,644 sentences), 294 parliamentary speeches (7,083 sentences) and 1,060 tweets.

4.3 Results

In the following, we discuss the performance of the classification approaches, their advantages and disadvantages as well as their applicability for the (comparative) study of different text types.

In general, the achieved accuracy scores (see Table 2) - especially for the best supervised model (BERT) - are comparable to similar multi-category topic classification tasks (e.g. Osnabrügge et al., 2021). The overall accuracy across all text types thereby shows a mostly expected distribution along the unsupervised - supervised axis. The completely unsupervised LDA underperforms significantly compared to the semi-supervised Newsmap, which in turn achieves lower accuracy than the supervised application of BERT. This is evident across all four tested types of text. In principle, supervised models, such as Naïve Bayes and BERT models, provide the inherent advantage that they are trained directly with the desired output format as target. This should, given enough training data and a sufficiently powerful model, theoretically always achieve higher accuracy for classification tasks compared to unsupervised and semi-supervised approaches. Large pre-trained language models (like BERT in our case) carry additional advantages. The pre-training on large volumes of text provides semantic and syntactic feature-rich embeddings of text. For example, they have the ability to map synonyms and complex relationships between words.

The results also show that the two supervised approaches (Naïve Bayes and BERT) perform significantly better when applied within-domain (manifestos) than cross-domain (press releases, parliamentary speeches, tweets). This follows our expectations as the studied text types differ on multiple dimensions. The supervised models learn the format and style of manifestos and are therefore much better-suited to classifying manifestos than the other types of text. As press releases are more similar in style to manifestos, it is not surprising that the results for them are better than for the parliamentary speeches and tweets.

It is also worth noting that the semi-supervised Newsmap performs better cross-domain than the supervised Naïve Bayes approach. This shows that, depending on the capabilities of the chosen model and the task at hand, supervised approaches are not necessarily better than semi-supervised approaches. In our application, only the state-of-the-art language model BERT is able to outperform the simpler semi-supervised approach in cross-domain topic classification. Furthermore, the performance of Newsmap stays comparatively stable across the different types of text in our application. This is, however, strongly dependent on the quality of the seed-word dictionary and how well the seed-words capture important topic keywords in the target texts. Here, profound domain-specific knowledge is necessary to develop a suitable seed-word dictionary that works for the respective text type under investigation. This seems to work quite well in our application, but can be problematic in others, especially when texts differ strongly with regard to the used vocabulary and style.

Table 2 ACCURACIES OF DIFFERENT TOPIC CLASSIFICATION MODELS

Type	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naïve Bayes	Naïve Bayes (Smoothing)	BERT	BERT (Smoothing)
Manifestos	0.18	0.24	0.36	0.41	0.44	0.46	0.56	0.56
Press releases	0.15	0.19	0.33	0.40	0.28	0.34	0.39	0.47
Parl. Speeches	0.17	0.21	0.33	0.37	0.24	0.29	0.34	0.41
Tweets	0.21	-	0.32	-	0.24	-	0.39	-

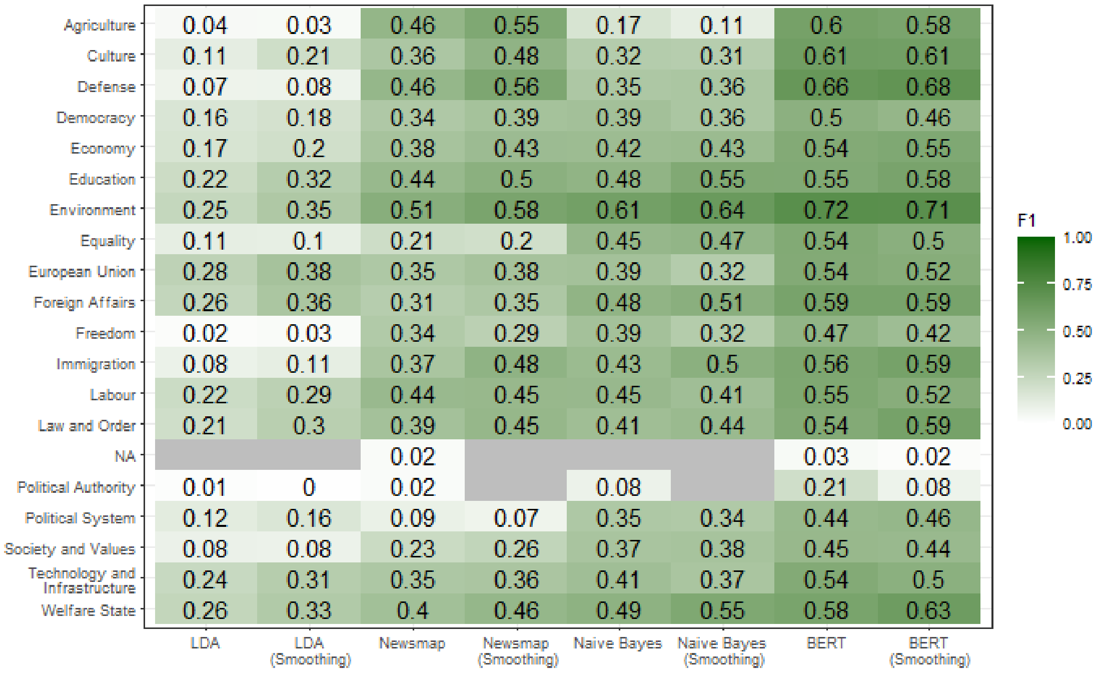
The results also clearly demonstrate the value of contextual smoothing in sentence-level classification tasks. In almost all cases, the application of contextual smoothing achieves moderate to significant improvement in accuracy.[12] Updating individual sentence topic predictions by using surrounding sentence predictions seems to denoise and improve the individual sentence predictions. Interestingly, this effect is least pronounced for supervised learning in the case of manifestos. Two main reasons play a role here. Firstly, manifestos are carefully drafted over a long period of time and offer dense policy-related information in nearly every sentence. Thus, nearly every sentence contains policy-related content and information from surrounding sentences is not as necessary for classifying individual sentences than with other text types. Secondly, the supervised models (Naïve Bayes, BERT) are trained on manifestos in our application and therefore well-suited for classifying this type of text.

A look at the classification performance in the individual categories shows that BERT also achieves the best result here overall, but Newsmap and Naïve Bayes are on a par in some categories (see Figure 3 - Figure

[12] We did not use contextual smoothing the case of tweets as we classified them at document-level.

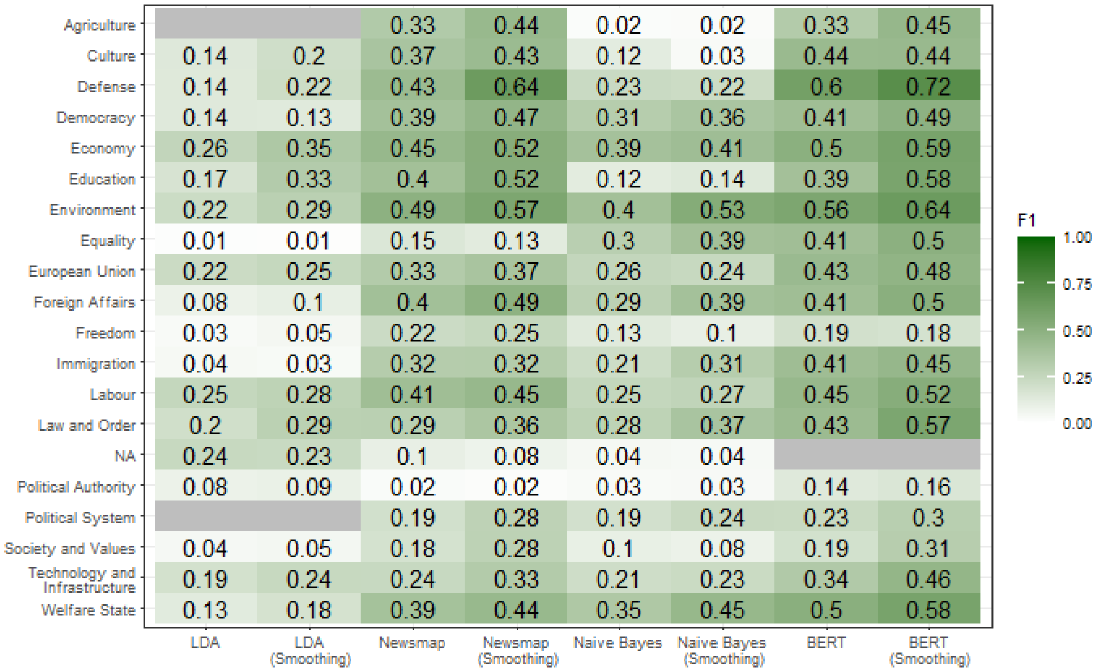
6). Newsmap for example achieves similar performance levels as BERT when it comes to the topics of “Agriculture” and “Education”. Again, the quality of seed-word selection is highly important here. Users need high levels of domain-specific knowledge to identify suitable seed-words that reliably capture topics and work across different types of text. The performance of LDA, however, is clearly falling behind and is virtually non-existent in some categories. This is even the case in categories, where the other approaches achieve a comparatively good performance (e.g. “Foreign Affairs”). This illustrates well the aforementioned problem of matching desired and actual topic output and the difficulty of their subsequent interpretation and post-processing. We were for instance often unable to match every topic from our codebook with a topic delivered by the LDA model. This is indicated by grey cells in the respective figures (Figure 3 - Figure 6). However, LDA achieves very good results for the additional category “Greeting”, which we introduced in the case of parliamentary speeches (Figure 5). Here, LDA´s strength in exploration and ability to detect (unexpected) text features is highly valuable. This can be important when working with multiple different text types.

Figure 3 COMPARISON OF F1-SCORES FOR MANIFESTOS



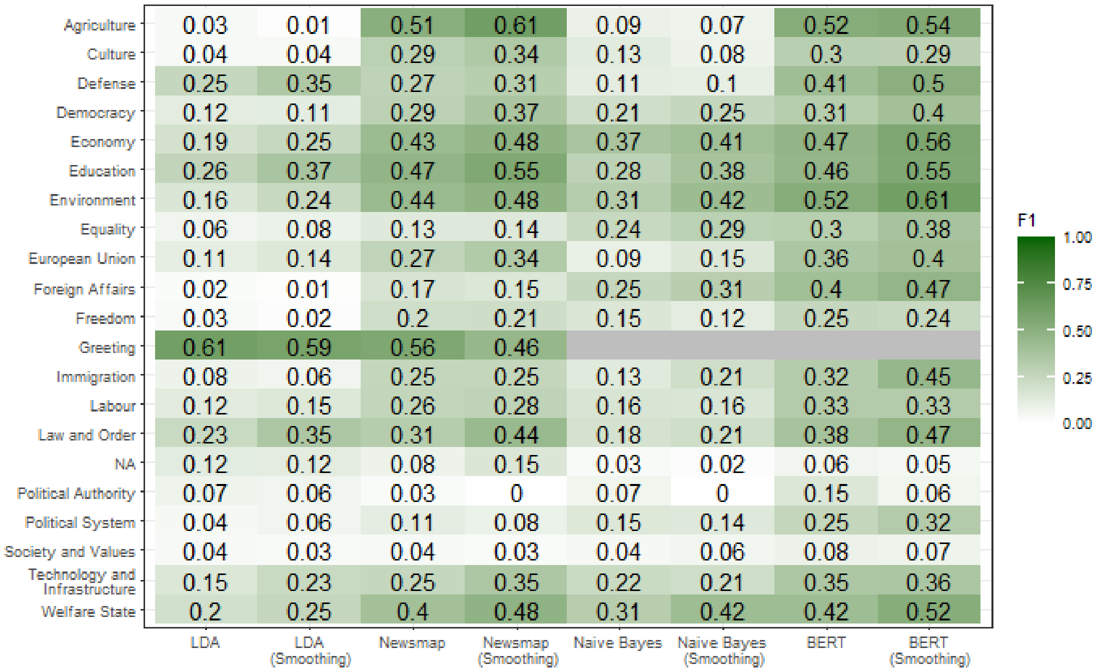
	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naive Bayes	Naive Bayes (Smoothing)	BERT	BERT (Smoothing)
Agriculture	0.04	0.03	0.46	0.55	0.17	0.11	0.6	0.58
Culture	0.11	0.21	0.36	0.48	0.32	0.31	0.61	0.61
Defense	0.07	0.08	0.46	0.56	0.35	0.36	0.66	0.68
Democracy	0.16	0.18	0.34	0.39	0.39	0.36	0.5	0.46
Economy	0.17	0.2	0.38	0.43	0.42	0.43	0.54	0.55
Education	0.22	0.32	0.44	0.5	0.48	0.55	0.55	0.58
Environment	0.25	0.35	0.51	0.58	0.61	0.64	0.72	0.71
Equality	0.11	0.1	0.21	0.2	0.45	0.47	0.54	0.5
European Union	0.28	0.38	0.35	0.38	0.39	0.32	0.54	0.52
Foreign Affairs	0.26	0.36	0.31	0.35	0.48	0.51	0.59	0.59
Freedom	0.02	0.03	0.34	0.29	0.39	0.32	0.47	0.42
Immigration	0.08	0.11	0.37	0.48	0.43	0.5	0.56	0.59
Labour	0.22	0.29	0.44	0.45	0.45	0.41	0.55	0.52
Law and Order	0.21	0.3	0.39	0.45	0.41	0.44	0.54	0.59
NA			0.02				0.03	0.02
Political Authority	0.01	0	0.02		0.08		0.21	0.08
Political System	0.12	0.16	0.09	0.07	0.35	0.34	0.44	0.46
Society and Values	0.08	0.08	0.23	0.26	0.37	0.38	0.45	0.44
Technology and Infrastructure	0.24	0.31	0.35	0.36	0.41	0.37	0.54	0.5
Welfare State	0.26	0.33	0.4	0.46	0.49	0.55	0.58	0.63

Figure 4 COMPARISON OF F1-SCORES FOR PRESS RELEASES



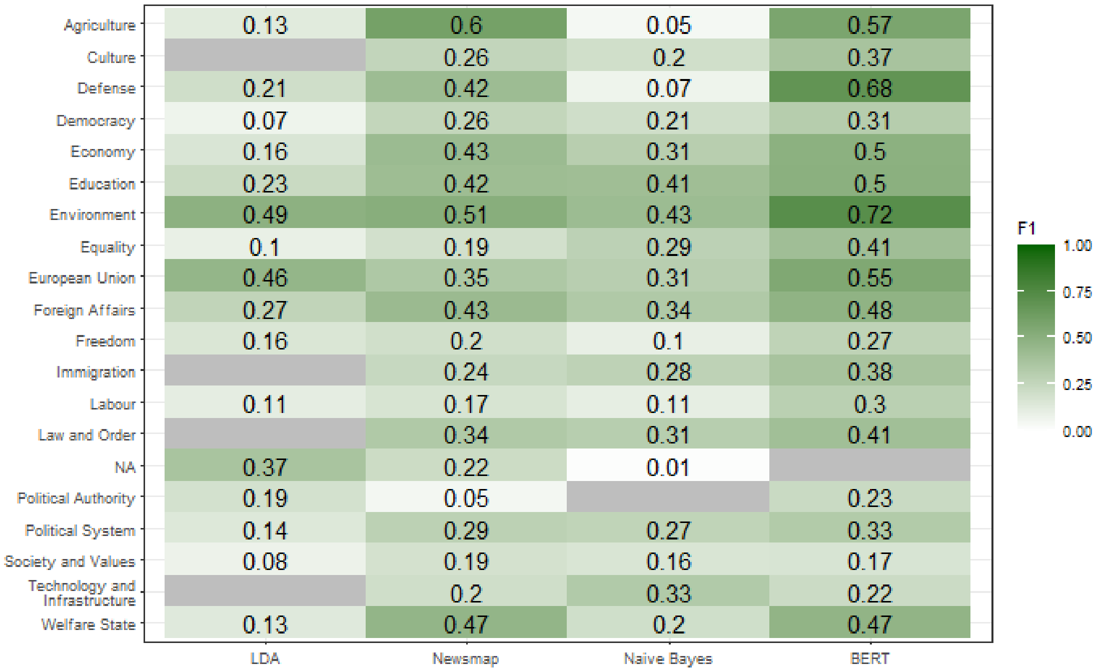
	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naive Bayes	Naive Bayes (Smoothing)	BERT	BERT (Smoothing)
Agriculture			0.33	0.44	0.02	0.02	0.33	0.45
Culture	0.14	0.2	0.37	0.43	0.12	0.03	0.44	0.44
Defense	0.14	0.22	0.43	0.64	0.23	0.22	0.6	0.72
Democracy	0.14	0.13	0.39	0.47	0.31	0.36	0.41	0.49
Economy	0.26	0.35	0.45	0.52	0.39	0.41	0.5	0.59
Education	0.17	0.33	0.4	0.52	0.12	0.14	0.39	0.58
Environment	0.22	0.29	0.49	0.57	0.4	0.53	0.56	0.64
Equality	0.01	0.01	0.15	0.13	0.3	0.39	0.41	0.5
European Union	0.22	0.25	0.33	0.37	0.26	0.24	0.43	0.48
Foreign Affairs	0.08	0.1	0.4	0.49	0.29	0.39	0.41	0.5
Freedom	0.03	0.05	0.22	0.25	0.13	0.1	0.19	0.18
Immigration	0.04	0.03	0.32	0.32	0.21	0.31	0.41	0.45
Labour	0.25	0.28	0.41	0.45	0.25	0.27	0.45	0.52
Law and Order	0.2	0.29	0.29	0.36	0.28	0.37	0.43	0.57
NA	0.24	0.23	0.1	0.08	0.04	0.04		
Political Authority	0.08	0.09	0.02	0.02	0.03	0.03	0.14	0.16
Political System			0.19	0.28	0.19	0.24	0.23	0.3
Society and Values	0.04	0.05	0.18	0.28	0.1	0.08	0.19	0.31
Technology and Infrastructure	0.19	0.24	0.24	0.33	0.21	0.23	0.34	0.46
Welfare State	0.13	0.18	0.39	0.44	0.35	0.45	0.5	0.58

Figure 5 COMPARISON OF F1-SCORES FOR PARLIAMENTARY SPEECHES



	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naive Bayes	Naive Bayes (Smoothing)	BERT	BERT (Smoothing)
Agriculture	0.03	0.01	0.51	0.61	0.09	0.07	0.52	0.54
Culture	0.04	0.04	0.29	0.34	0.13	0.08	0.3	0.29
Defense	0.25	0.35	0.27	0.31	0.11	0.1	0.41	0.5
Democracy	0.12	0.11	0.29	0.37	0.21	0.25	0.31	0.4
Economy	0.19	0.25	0.43	0.48	0.37	0.41	0.47	0.56
Education	0.26	0.37	0.47	0.55	0.28	0.38	0.46	0.55
Environment	0.16	0.24	0.44	0.48	0.31	0.42	0.52	0.61
Equality	0.06	0.08	0.13	0.14	0.24	0.29	0.3	0.38
European Union	0.11	0.14	0.27	0.34	0.09	0.15	0.36	0.4
Foreign Affairs	0.02	0.01	0.17	0.15	0.25	0.31	0.4	0.47
Freedom	0.03	0.02	0.2	0.21	0.15	0.12	0.25	0.24
Greeting	0.61	0.59	0.56	0.46				
Immigration	0.08	0.06	0.25	0.25	0.13	0.21	0.32	0.45
Labour	0.12	0.15	0.26	0.28	0.16	0.16	0.33	0.33
Law and Order	0.23	0.35	0.31	0.44	0.18	0.21	0.38	0.47
NA	0.12	0.12	0.08	0.15	0.03	0.02	0.06	0.05
Political Authority	0.07	0.06	0.03	0	0.07	0	0.15	0.06
Political System	0.04	0.06	0.11	0.08	0.15	0.14	0.25	0.32
Society and Values	0.04	0.03	0.04	0.03	0.04	0.06	0.08	0.07
Technology and Infrastructure	0.15	0.23	0.25	0.35	0.22	0.21	0.35	0.36
Welfare State	0.2	0.25	0.4	0.48	0.31	0.42	0.42	0.52

Figure 6 COMPARISON OF F1-SCORES FOR TWEETS



The role of the "NA" category is also worth a special note. Designed simply as a general residual category for all sentences that cannot be classified into a category, it poses a problem for the classification models due to its text type dependency. How NAs can be systematically described and coded and which keywords characterize them differs significantly across text types. For instance, while press releases can consist exclusively of non-policy-related messages (e.g. event information, organisational matters), parliamentary speeches, with their transcribed spoken word, tend to contain many domain-specific phrases such as greetings for example. Unsupervised and semi-supervised models are much more adaptive in this regard than supervised ones. In the case of semi-supervised models (e.g. Newsmap), an additional category for further keywords can easily be added to the seed-word dictionary. A similar adaptation is also possible in the case of unsupervised topic models (e.g. LDA). Here, a new category can simply be added to the codebook during transfer topic labelling. We made use of both adaptations during the application of LDA and Newsmap to parliamentary speeches by adding the topic category "Greeting" (see additional topic category in Figure 5). This is not possible and scalable in our application of supervised learning as additional labelled training data, which includes the category "Greeting", would be necessary. To mitigate this problem, a two-stage classification could be useful, in which policy-related sentences are first distinguished from non-policy-related sentences (e.g. greetings) before the actual topic classification begins.

5. Conclusion

The social sciences have made significant progress when it comes to the application of computer-based methods to study political texts. A wealth of tools and studies exist, but significant gaps remain. Systematic comparisons of different approaches and tools as well as their applicability to study various types of political text are largely missing. The (comparative) analysis of multiple text types produced by political actors is, however, a highly relevant field and computer-based text analysis a promising way of exploring this field. By using computer-based tools, researchers are for example able to compare communication strategies of political actors in different communication channels or track the evolution of policy preferences at different stages of the political process (i.e. in manifestos, speeches, social media, coalition agreements, legislative texts). Such analyses are becoming increasingly important in order to keep up with political processes. Especially in the age of digitalization and social media, political processes take place across different spheres (parliamentary arena, social media etc.) and at increasing velocity. Therefore, it is necessary to capture political processes in their entirety and not only small snapshots.

This deliverable makes three contributions to move the (comparative) study of different political texts forward. Firstly, we provide an overview of widely-used computer-based text analysis methods in the social sciences. We discuss approaches to topic classification, scaling and sentiment analysis. Secondly, we identify potential challenges arising from the (comparative) study of different text types. We show that political texts differ on a number of dimensions (format, style, conventions, purpose) and how this can affect computer-based text analysis. Thirdly, we present a case study that evaluates the applicability of unsupervised, semi-supervised and supervised topic classification tools to the (comparative) study of different types of political text (manifestos, press releases, parliamentary speeches and tweets). The results demonstrate that text analysis can work across different text types, but performance varies depending on the applied tool and studied text type.

Overall, the deliverable shows that users need to be aware of two connected aspects when applying computer-based text analysis. On the one hand, different approaches and tools offer specific advantages and disadvantages, such as the required domain-specific knowledge or the necessary resources. On the other hand, text characteristics (e.g. format, style, conventions, purpose) can influence the applicability of computer-based text analysis methods. Users need to be aware of method and text characteristics to identify potential problems and develop adequate solutions.

Here, the OPTED infrastructure will provide users with valuable assistance for the (comparative) study of different political texts. Firstly, the OPTED platform will identify and point towards relevant textual data sets and tools (for previous steps, see: D2.1, D3.1, D4.2, D5.1, D6.1). Users will be able to search the database for existing resources (text corpora, tools) and retrieve information about them (e.g. coverage, quality, availability). This will not only help users to find texts as subjects for further analysis, but also, for example, to identify suitable annotated text corpora as training data for supervised text analysis models. Secondly, OPTED develops routines and guidelines for data pre-processing, storage and sharing (e.g. D7.1, D7.2) as well as harmonization and linkage (e.g. D8.1). Our deliverable shows that different types of political texts and methods require different steps in the text analysis workflow. Developing guidelines and tools that, for example, focus on the handling of symbols (e.g. emojis) or specific conventions (e.g. direct quotes) during pre-processing is crucial to move the (comparative) study of political texts forward. The OPTED infrastructure will provide users with valuable support in this field.

References

APA-OTS. (n.d.). https://www.ots.at/

Baden, C., Pipal, C., Schoonvelde, M., and van der Velden, M. A. C. G. (2022). Three Gaps in Computational Text Analysis Methods for Social Sciences: A Research Agenda. *Communication Methods and Measures*, 16 (1), 1–18. https://doi.org/10.1080/19312458.2021.2015574

Banda, Kevin K. (2015). Competition and the dynamics of issue convergence. *American Politics Research*, 43 (5), 821–845. https://doi.org/10.1177/1532673X14564570

Banerjee, S., Ramanathan, K., and Gupta, A. (2007). Clustering short texts using wikipedia. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 787–788. https://doi.org/10.1145/1277741.1277909

Barberá, P., Casas, A., Nagler, J., Egan, P. J., Bonneau, R., Jost, J. T., and Tucker, J. A. (2019). Who Leads? Who Follows? Measuring Issue Attention and Agenda Setting by Legislators and the Mass Public Using Social Media Data. *American Political Science Review*, 113 (4), 883–901. https://doi.org/10.1017/S0003055419000352

Béchara, H., Herzog, A., Jankin, S., and John, P. (2021). Transfer learning for topic labelling: Analysis of the UK House of Commons speeches 1935–2014. *Research & Politics*, 8 (2), 20531680211022206. https://doi.org/10.1177/20531680211022206

Benoit. (2020). Text as data: An overview. In L. Curini and R. Franzese (Eds.), *The SAGE Handbook of Research Methods in Political Science and International Relations*. Sage Publishing.

Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., and Matsuo, A. (2018). Quanteda: An R package for the quantitative analysis of textual data. *Journal of Open Source Software*, 3 (30), 774. https://doi.org/10.21105/joss.00774

Benoit, K., Watanabe, K., Wang, H., Perry, P. O., Lauderdale, B., Gruber, J., and Lowe, W. (2021). *Quanteda.textmodels: Scaling Models and Classifiers for Textual Data*. https://CRAN.R-project.org/package=quanteda.textmodels

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, 993–1002.

Budge, I., Robertson, D., and Hearl, D. (1987). Ideology, Strategy and Party Change: Spatial Analyses of Post-War Election Programmes in 19 Democracies. Cambridge University Press.

Burscher, B., Vliegenthart, R., and De Vreese, C. H. (2015). Using Supervised Machine Learning to Code Policy Issues: Can Classifiers Generalize across Contexts? *The ANNALS of the American Academy of Political and Social Science*, 659 (1), 122–131. https://doi.org/10.1177/0002716215569441

Burst, T., Krause, W., Lehmann, P., Lewandowski, J., Matthieß, T., Merz, N., Regel, S. & Zehnter, L. (2021). *Manifesto Corpus*. Version: 2021-1. Berlin. WZB Berlin Social Science Center.

Chan, C.-H., Zeng, J., Wessler, H., Jungblut, M., Welbers, K., Bajjalieh, J. W., Atteveldt, W. van, and Althaus, S. L. (2020). Reproducible Extraction of Cross-lingual Topics. *Communication Methods and Measures*, 14 (4), 285–305. https://doi.org/10.1080/19312458.2020.1812555

Chang, C., and Masterson, M. (2020). Using Word Order in Political Text Classification with Long Short-term Memory Models. *Political Analysis*, 28 (3), 395–411. https://doi.org/10.1017/pan.2019.46

Chapelle, O., Schölkopf, B., Zien, A., and Bach, F. (Eds.). (2006). *Semi-Supervised Learning*. MIT Press.

Dalmus, C., Hänggli, R., and Bernhard, L. (2017). The charm of salient issues? Parties' strategic behavior in press releases. In P. Van Aelst and S. Walgrave (Eds.), *How Political Actors Use the Media* (pp. 187–205). Springer.

Damore, D. F. (2005). Issue Convergence in Presidential Campaigns. *Political Behavior*, 27 (1), 71–97.

Debus, M., and Tosun, J. (2021). The manifestation of the green agenda: A comparative analysis of parliamentary debates. *Environmental Politics*, 30 (6), 918–937. https://doi.org/10.1080/09644016.2020. 1864130

Denny, M., & Spirling, A. (2018). Text Preprocessing For Unsupervised Learning: Why It Matters, When It Misleads, And What To Do About It. *Political Analysis*, *26*(2), 168-189. https://doi.org/10.1017/pan.2017.44

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [Cs]*. http://arxiv.org/abs/1810.04805

Downs, A. (1957). An Economic Theory of Democracy. Harper & Brothers.

Elmelund-Præstekær, C. (2011). Mapping Parties' Issue Agenda in Different Channels of Campaign Communication: A Wild Goose Chase? *Javnost-the public*, 18 (1): 37–51.

Gilardi, F., Baumgartner, L., Dermont, C., Donnay, K., Gessler, T., Kubli, M., Leemann, L., and Müller, S. (2022). Building Research Infrastructures to Study Digital Technology and 16 Politics: Lessons from Switzerland. *PS: Political Science & Politics*, 55 (2), 354–359. https://doi.org/10.1017/S1049096521000895

Gilardi, F., Gessler, T., Kubli, M., and Müller, S. (2022). Social Media and Political Agenda Setting. *Political Communication*, 39 (1), 39–60. https://doi.org/10.1080/10584609.2021.1910390

González-Bailón, S., & Paltoglou, G. (2015). Signals of Public Opinion in Online Communication: A Comparison of Methods and Data Sources. *The ANNALS of the American Academy of Political and Social Science*, 659 (1), 95–107. https://doi.org/10.1177/0002716215569192

Green, J., and Hobolt, S. B. (2008). Owning the issue agenda: Party strategies and vote choices in British elections. *Electoral Studies*, 27 (3), 460–476. https://doi.org/10.1016/j.electstud.2008.02.003

Green-Pedersen, C. (2007). The Growing Importance of Issue Competition: The Changing Nature of Party Competition in Western Europe. *Political Studies*, 55 (3), 607–628. https://doi.org/10.1111/j.1467-9248.2007.00686.x

Grimmer, J., & Stewart, B. (2013). Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts. *Political Analysis*, *21*(3), 267-297. https://doi.org/10.1093/pan/mps028

Guinaudeau, I., and Persico, S. (2014). What is Issue Competition? Conflict, Consensus and Issue Ownership in Party Competition. *Journal of Elections, Public Opinion and Parties*, 24 (3), 312–333. https://doi.org/10.1080/17457289.2013.858344

Haselmayer, M. (2019). Negative campaigning and its consequences: A review and a look ahead. *French Politics*, 17, 355–372. https://doi.org/10.1057/s41253-019-00084-8

Haselmayer, M., Jenny, M. (2017). Sentiment analysis of political communication: Combining a dictionary approach with crowdcoding. *Quality & Quantitiy* 51**,** 2623–2646. https://doi.org/10.1007/s11135-016-0412-4

Haselmayer, M., Wagner, M., and Meyer, T. M. (2017). Partisan Bias in Message Selection: Media Gatekeeping of Party Press Releases. *Political Communication*, 34 (3), 367–384. https://doi.org/10.1080/10584609.2016.1265619

Hopmann, D. N., Elmelund-Præstekær, C., Albæk, E., Vliegenthart, R., and Vreese, C. H. de. (2012). Party media agenda-setting: How parties influence election news coverage. *Party Politics*, 18 (2), 173– 191. https://doi.org/10.1177/1354068810380097

Jerzak, C. T., King, G., and Strezhnev, A. (2022). An Improved Method of Automated Nonparametric Content Analysis for Social Science. *Political Analysis*, 1–17. https://doi.org/10.1017/pan.2021.36

Laver, M., Benoit, K., and Garry, J. (2003). Extracting Policy Positions from Political Texts Using Words as Data. *American Political Science Review*, *97*(2), 311-331. https://doi.org/10.1017/S0003055403000698

Maier, D., Baden, C., Stoltenberg, D., De Vries-Kedem, M., and Waldherr, A. (2022). Machine Translation Vs. Multilingual Dictionaries Assessing Two Strategies for the Topic Modeling of Multilingual Text Collections. *Communication Methods and Measures*, 16 (1), 19-38. https://doi.org/10.1080/19312458.2021.1955845

Maier, D., Waldherr, A., Miltner, P., Wiedemann, G., Niekler, A., Keinert, A., Pfetsch, B., Heyer, G., Reber, U., Häussler, T., Schmid-Petri, H., and Adam, S. (2018). Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology. *Communication Methods and Measures*, 12 (2-3), 93–118. https://doi.org/10.1080/19312458.2018.1430754

Meyer, T. M., Haselmayer, M., and Wagner, M. (2020). Who Gets into the Papers? Party Campaign Messages and the Media. *British Journal of Political Science*, 50 (1), 281–302. https://doi.org/10.1017/S0007123417000400

Norris, P. (1999). On Message: Communicating the Campaign. SAGE.

Osnabrügge, M., Ash, E., and Morelli, M. (2021). Cross-Domain Topic Classification for Political Texts. *Political Analysis*, 1–22. https://doi.org/10.1017/pan.2021.37

Petkevic, V., & Nai, A. (2022). Political Attacks in 280 Characters or Less: A New Tool for the Automated Classification of Campaign Negativity on Social Media. *American Politics Research*, 50(3), 279–302. https://doi.org/10.1177/1532673X211055676

Phan, X.-H., Nguyen, L.-M., and Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. Proceedings of the 17th International Conference on World Wide Web, 91–100. https://doi.org/10.1145/1367497.1367510

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. https://doi.org/10.1111/j.1540-5907.2009.00427.x

Rauh, C., and Schwalbach, J. (2020). *The ParlSpeech V2 data set: Full-text corpora of 6.3 million parliamentary speeches in the key legislative chambers of nine representative democracies*. Harvard Dataverse. https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/L4OAKN

Roberts, M. E., Stewart, B. M., and Airoldi, E. M. (2016). A Model of Text for Experimentation in the Social Sciences. *Journal of the American Statistical Association*, 111 (515), 988–1003. https://doi.org/10.1080/01621459.2016.1141684

Robertson, D. B. (1976). A theory of party competition. John Wiley & Sons.

Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Emrich, Š., and Sedlmair, M. (2018). More than Bags of Words: Sentiment Analysis with Word Embeddings. *Communication Methods and Measures*, 12 (2-3), 140-157. https://doi.org/10.1080/19312458.2018.1455817

Schönhofen, P. (2009). Identifying document topics using the Wikipedia category network. *Web Intelligence and Agent Systems: An International Journal*, 7 (2), 195–207. https://doi.org/10.3233/WIA-2009-0162

Sigelman, L., and Buell, E.H., Jr. (2004). Avoidance or engagement? Issue convergence in US presidential campaigns, 1960–2000. *American Journal of Political Science*, 48 (4), 650–661. https://doi.org/10.1111/j.0092-5853.2004.00093.x

Slapin, J., and Proksch, S-O. (2008). A scaling model for estimating time-series party positions from texts. *American Journal of Political Science*, 52(3): 705–22. https://doi.org/10.1111/j.1540-5907.2008.00338.x

Song, H., Tolochko, P., Eberl, J.-M., Eisele, O., Greussing, E., Heidenreich, T., Lind, F., Galyga, S., and Boomgaarden, H. G. (2020). In Validations We Trust? The Impact of Imperfect Human Annotations as a Gold Standard on the Quality of Validation of Automated Content Analysis. *Political Communication*, 37 (4), 550–572. https://doi.org/10.1080/10584609.2020.1723752

Tresch, A., Lefevere, J., and Walgrave, S. (2018). How parties' issue emphasis strategies vary across communication channels: The 2009 regional election campaign in Belgium. *Acta Politica*, 53 (1), 25–47. https://doi.org/10.1057/s41269-016-0036-7

Valentim, V., Widmann, T. (2021). Does Radical-Right Success Make the Political Debate More Negative? Evidence from Emotional Rhetoric in German State Parliaments. *Political Behavior*. https://doi.org/10.1007/s11109-021-09697-8

Van Atteveldt, W., Kleinnijenhuis, J., Ruigrok, N., & Schlobach, S. (2008). Good News or Bad News? Conducting Sentiment Analysis on Dutch Text to Distinguish Between Positive and Negative Relations. *Journal of Information Technology & Politics*, 5 (1), 73-94. https://doi.org/10.1080/19331680802154145

Volkens, A., Burst, T., Krause, W., Lehmann, P., Matthieß, T., Regel, S., Weßels, B., Zehnter, L., and Sozialforschung (WZB), W. B. F. (2021). *Manifesto Project Dataset*. Manifesto Project. https://manifesto-project.wzb.eu/doi/manifesto.mpds.2021a

Watanabe, K. (2018). Newsmap. Digital Journalism, 6 (3), 294–309. https://doi.org/10.1080/21670811.2017.1293487

Watanabe, K., and Zhou, Y. (2022). Theory-Driven Analysis of Large Corpora: Semisupervised Topic Classification of the UN Speeches. *Social Science Computer Review*, 40 (2), 346–366. https://doi.org/10. 1177/0894439320907027

Welbers, K., Van Atteveldt, W., and Benoit, K. (2017). Text Analysis in R. *Communication Methods and Measures*, 11 (4), 245–265. https://doi.org/10.1080/19312458.2017.1387238

Werner, A., Lacewell, O., Volkens, A., Matthieß, T., Zehnter, L. & van Rinsum, L. (2021). *Manifesto Coding Instructions (5th re-revised edition)*. Manifesto Project.

Widmann, T., & Wich, M. (2022). Creating and Comparing Dictionary, Word Embedding, and Transformer-Based Models to Measure Discrete Emotions in German Political Text. *Political Analysis*, 1-16. https://doi.org/10.1017/pan.2022.15

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Platen, P. von, Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., . . . Rush, A. M. (2020). HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771 [Cs]*. http://arxiv.org/abs/1910.03771

Young, L., & Soroka, S. (2012), Affective News: The Automated Coding of Sentiment in Political Texts. *Political Communication*, 29:2, 205-231. https://doi.org/10.1080/10584609.2012.671234

Zelikovitz, S., and Hirsh, H. (2000). Improving Short Text Classification Using Unlabelled Background Knowledge. *Proceedings of the 17th International Conference on Machine Learning*, Stanford, CA, USA. https://www.semanticscholar.org/paper/Improving-Short-Text-Classification-Using-Unlabelled-Zelikovitz-Hirsh/488517c595b07acaa8a4f3b33eb7dac5fc185766

Zumbach, D. (2020). *Swissparl: Interface to the Webservices of the Swiss Parliament*. R package version 0.2.1. https://CRAN.R-project.org/package=swissparl

Appendix

Appendix A

Table A1 displays the codebook used for this study. The codebook is based on the Manifesto Project (MARPOR) coding scheme. The individual MAPROR codes are assigned to broader issue categories for this study.[13]

Table A1 **Adapted codebook based on Manifesto Project (MARPOR) coding scheme.**

Issue	MARPOR codes
Agriculture	703.1, 703.2
Culture	502
Defense	104, 105
Democracy	202.1, 202.2, 202.3, 202.4, 203, 204
Economy	401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 412, 413, 414, 415, 704, 416.1
Education	506, 507
Environment	416.2, 501
Equality	503, 705
European Union	108, 110
Foreign Affairs	101, 102, 103.1, 103.2, 106, 107, 109
Freedom	201.1, 201.2
Immigration	601.2, 602.2, 607.1, 607.2, 607.3, 608.1, 608.2, 608.3
Labour	701, 702
Law and Order	605.1, 605.2
NA	000
Political Authority	305.1, 305.2, 305.3, 305.4, 305.5, 305.6
Political System	301, 302, 303, 304
Society and Values	601.1, 602.1, 603, 604, 606.1, 606.2
Technology and Infrastructure	411
Welfare State	504, 505, 706

[13] In the case of parliamentary speeches we have added “Greeting” as a further category to capture greetings and direct addresses, which occur frequently in parliamentary speeches.

Appendix B

The following sections provide detailed descriptions of our LDA, Newsmap, Naive Bayes and BERT applications.

LDA

In the case of LDA, we run multiple configurations to identify the best-fitting model. To implement LDA we mainly rely on the R package *quanteda* (Benoit et al., 2018). The following paragraphs discuss the workflow of the LDA application:

1. We perform several different steps with regard to pre-processing. Firstly, we engage in some limited domain-specific cleaning. Secondly, we split all German compound words. This is quite specific to our study as compound splitting is necessary to enable the automated transfer topic labelling during post-processing. The German language consists of multiple compound words. Splitting these words allows us to find better matches between words occuring in the text documents and words appearing in the codebook. Only this way, transfer topic labelling can work in our specific application. Thirdly, we remove all punctuation, numbers, symbols, URLs, separators and stopwords (e.g. articles, prepositions). Here, we use the available functions and stopword lists provided by *quanteda* (Benoit et al., 2018). Then we perform tokenization and stemming.
2. We run different configurations of LDA models per text type. This way we are able to identify the best-fitting model. Firstly, we specify different numbers of topics (*k*). We run models with k = 20, 25, 30, 40 and 50. Secondly, we calculate individual models per country and models for all three countries combined. Combining the different configurations allows us to identify the best-fitting LDA model.
3. During post-processing, we apply two processes. Firstly, we use automated transfer topic labelling following Béchara et al. (2021) to match the model output with the topic categories from our codebook. Secondly, we apply contextual smoothing to the model output (see: Watanabe and Zhou, 2022). Here, we update single sentence predictions by surrounding sentence predictions (window of 2 sentences before and after the respective sentence). Hence, we update the classification of each individual sentence by taking the context (surrounding) sentences into account.

Newsmap

Newsmap is a seeded-model and therefore requires a seed-word dictionary as input. To implement Newsmap we mainly rely on the R packages *quanteda* (Benoit et al., 2018) and *newsmap* (Watanabe, 2018). We apply Newsmap in four steps:

1. To implement Newsmap, we first create a seed-word dictionary. Here, we identify and include relevant keywords (seed-words) for each of the 20 topic categories defined in our codebook. To identify the seed-words, we follow the advice by Watanabe and Zhou (2022) and combine both

knowledge-based and frequency-based seed-words.

2. We perform several different steps with regard to pre-processing. Firstly, we engage in some limited domain-specific cleaning. Here, we remove some frequent words (e.g. party names) that might produce false hits by the dictionary. Secondly, we remove all punctuation, numbers, symbols, URLs, separators and stopwords (e.g. articles, prepositions). Here, we use the available functions and stopword lists provided by *quanteda* (Benoit et al., 2018). Thirdly, we perform tokenization. During tokenization, we also compound tokens that are present as compound words in the seed-word dictionary.
3. We apply the Newsmap textmodel to the respective text documents. The model first searches for seed-words from the dictionary in all documents and assigns topic categories to these documents. The model then estimates the association between the topics and textual features via the co-occurrence of words. Based on this estimation, Newsmap assigns topic labels to all documents (Watanabe, 2018; Watanabe and Zhou, 2022).
4. Similar to the classification with LDA, we apply contextual smoothing (see: Watanabe and Zhou, 2022).

Naive Bayes

Naive Bayes classifiers use supervised learning and are implemented in the R packages *quanteda* (Benoit et al., 2018) and *quanteda.textmodels* (Benoit et al., 2021). We apply Naive Bayes in four steps:

1. Similar to the Newsmap application, we perform different pre-processing steps. Firstly, we engage in some limited domain-specific cleaning. Secondly, we remove all punctuation, numbers, symbols, URLs, separators and stopwords (e.g. articles, prepositions). Here, we use the available functions and stopwordlists provided by *quanteda* (Benoit et al., 2018). Thirdly, we perform tokenization.
2. After pre-processing, we train the supervised Naive Bayes classifier. During this training, we expose the Naive Bayes model to already labelled data. Based on that, the model learns its classification task. In this application, we train the Naive Bayes model on labelled manifestos from Austria, Germany and Switzerland. We include manifestos published from 2013 onwards[14], but exclude the most recent ones as they are part of our corpus for classification. Overall, our labelled training data (Germany: 2017/09, Austria: 2013/09, 2017/10, Switzerland: 2015/10) consists of 32,627 (quasi-)sentences.
3. We apply the model trained on the labelled manifestos to the respective unlabelled target documents (manifestos, press releases, parliamentary speeches and tweets).
4. As in the case of LDA and Newsmap, we apply contextual smoothing (see: Watanabe and Zhou,

[14] In 2015 the Manifesto Project made some adjustments to the coding scheme. Therefore, we only include manifestos coded according to the 2015 coding scheme in our training data set. The manifestos of the 2013 Austrian election are also included because they were only coded in 2015 and therefore already according to the new coding scheme.

2022).

BERT

As a concrete BERT model, we use the pre-trained model weights of a mulitlingual BERT (Devlin et al., 2019) model in the cased variant. The pre-trained model weights were obtained and the training was performed using the *huggingface* package (Wolf et al., 2020). We apply BERT in the following steps:

1. We split the German, Austrian and Swiss Manifestos into two parts: Manifestos from older election years constitute the training data set (Germany: 2017/09, Austria: 2013/09, 2017/10, Switzerland: 2015/10, total 32,627 quasi-sentences). The most recent election years constitute the manifesto test data set (Germany: 2021/09, Austria: 2019/09, Switzerland: 2019/10, total 29,179 quasi-sentences) to report the final model performance.
2. We randomly split 20% of the training data to use it as validation data during training.
3. The quasi-records are transformed into token-ids by the BERT tokenizer without further pre-processing. A simple truncation and padding strategy is applied: Every text input is starting from its beginning truncated or padded to BERTs maximum token input length (512 tokens).
4. The training runs over 3 epochs with a batch size of 16 (without gradient accumulation). The AdamW Optimizer with a learning rate of 0.00005, a weight decay of 0.01 and 100 warm-up steps was chosen as an optimizer.
5. We apply the model trained on the labelled manifestos to the respective unlabelled target documents (manifestos, press releases, parliamentary speeches and tweets).
6. For Tweets we use the emoji Python package (https://pypi.org/project/emoji/) to convert emojis to textual representations with the demojize function.

Appendix C

To measure the performance of the applied computer-based text analysis models, we use human coding of manifestos, press releases, parliamentary speeches and tweets as a “gold standard”. In the case of manifestos, the manually coded “gold standard” is readily available via the Manifesto Project data set (Volkens et al., 2021). To create the “gold standard” for press releases, parliamentary speeches and tweets, we rely on trained student coders from the Manifesto Project. The coders use the original main codes from the Manifesto Project, which we then aggregate into the respective topic categories according to our adapted codebook (see Appendix A).

For creating the coding data set, we use stratified sampling of the texts to take potential differences between the covered countries, years and parties into account. In principle, we aimed to sample 1% of press releases, parliamentary speeches and tweets. We were able to achieve this threshold for press releases and tweets and even performed some oversampling to account for significant country differences. In the case of parliamentary speeches we were unable to sample 1% of speeches from each parliament. Our parliamentary speech data set consists of almost 1 million sentences, with speeches from the German Bundestag making up a large part of this. Coding 1% of the speeches (i.e. 10,000 sentences) would have exceeded our available resources. Hence, we sampled 1% of speeches from the Austrian and Swiss parliaments and 0.5% of speeches from the German Bundestag. Overall, our “gold standard” consists of 21 manifestos (29,179 sentences), 346 press releases (4,644 sentences), 294 parliamentary speeches (7,083 sentences) and 1,060 tweets.

The coding procedure of each text type followed three steps. Firstly, two student coders (main coder and additional coder) annotated a sub-sample of the texts (20% of the coding sample) as a pre-test. This allows us to calculate intercoder reliability (see: Table A2). Secondly, the main coder received feedback on the pre-test by an expert coding supervisor from the Manifesto Project. Thirdly, the main coder coded the complete sample to create the “gold standard” for the respective text type.

Table A2 INTERCODER AGREEMENT FOR MANUAL CODING

Type	MARPOR codes	Topics
Press releases	0.58	0.66
Parl. Speeches	0.56	0.59
Tweets	0.54	0.59

Appendix D

Figure A1 - Figure A4 compare the model performances (accuracy) for all calculated models and text types. In the case of manifestos (Figure A1), press releases (Figure A2) and parliamentary speeches (Figure A3) we applied the following models:

- 20 LDA models: different number of topics (*k*); individual models per country/combined model for all countries; with/without contextual smoothing
- 2 Newsmap models: with/without contextual smoothing
- 2 Naive Bayes models: with/without contextual smoothing
- 2 BERT models: with/without contextual smoothing

In the case of tweets (Figure A4), we applied slightly different models for two main reasons. Firstly, contextual smoothing is not applicable to tweets. We performed the classification at document-level as tweets are too short to use a different classification unit (i.e. sentence-level). Secondly, we used artificially prolonged tweets to calculate some LDA models. Following Maier et al. (2022) we temporarily concatenated five tweets authored by the same user and generated the topic model based on these artificially prolonged documents. The topic model was then applied to the original Twitter corpus. These adaptations result in the following models:

- 20 LDA models: different number of topics (*k*); individual models per country/combined model for all countries; with/without artificially prolonged texts
- Newsmap model
- 1 Naive Bayes model
- 1 BERT model

Figure A1 COMPARISON OF ALL MODEL PERFORMANCES (ACCURACY) FOR MANIFESTOS

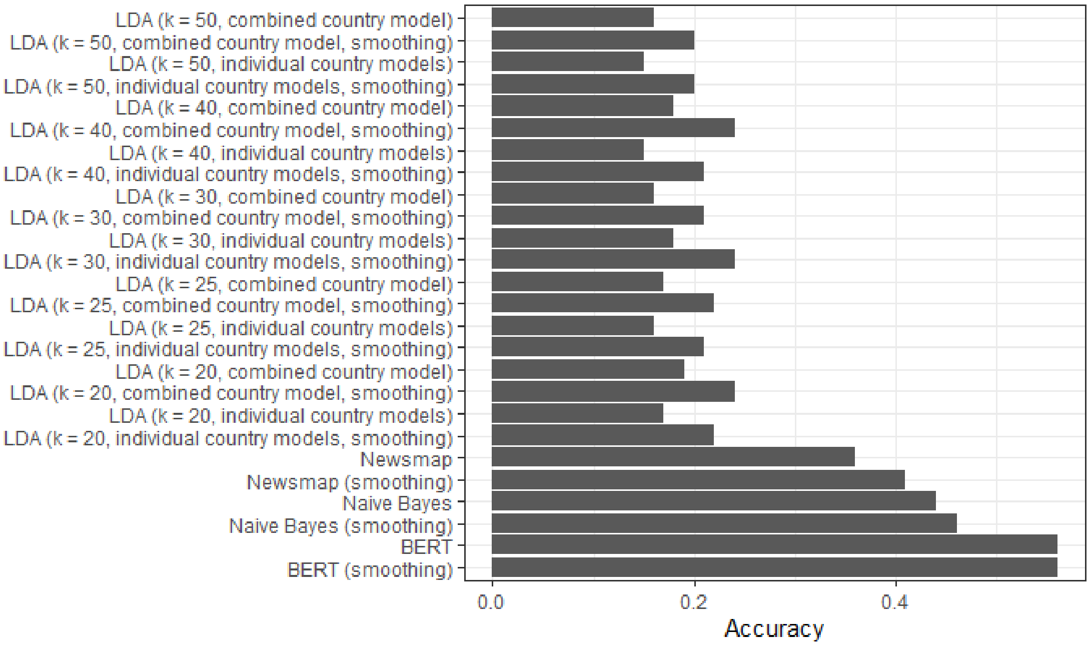


Figure A2 COMPARISON OF ALL MODEL PERFORMANCES (ACCURACY) FOR PRESS RELEASES

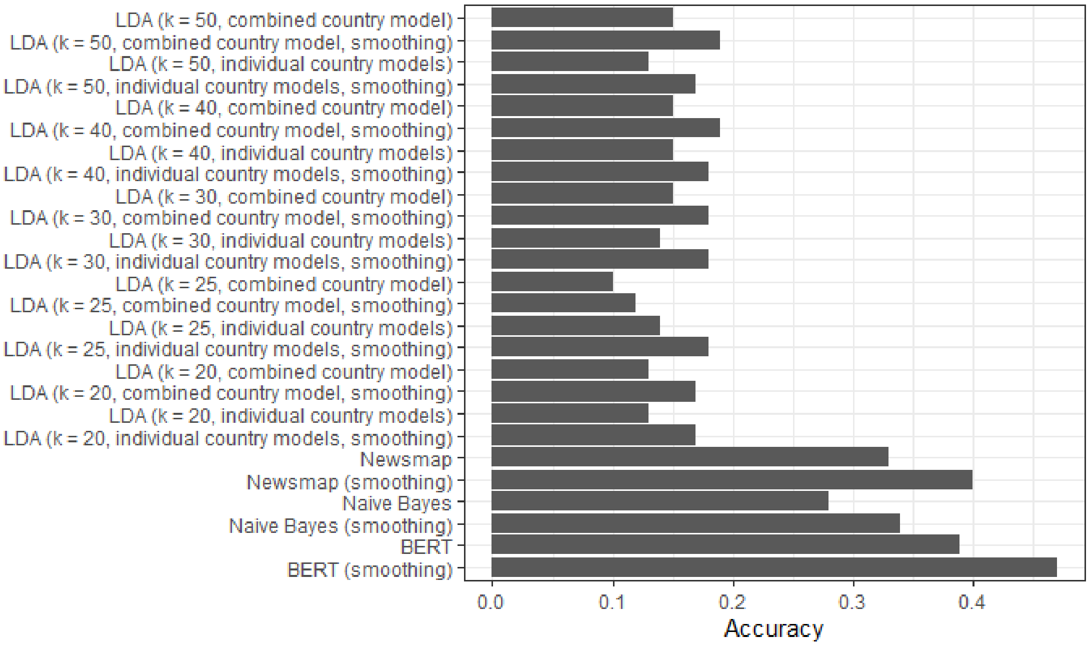


Figure A3 COMPARISON OF ALL MODEL PERFORMANCES (ACCURACY) FOR PARLIAMENTARY SPEECHES

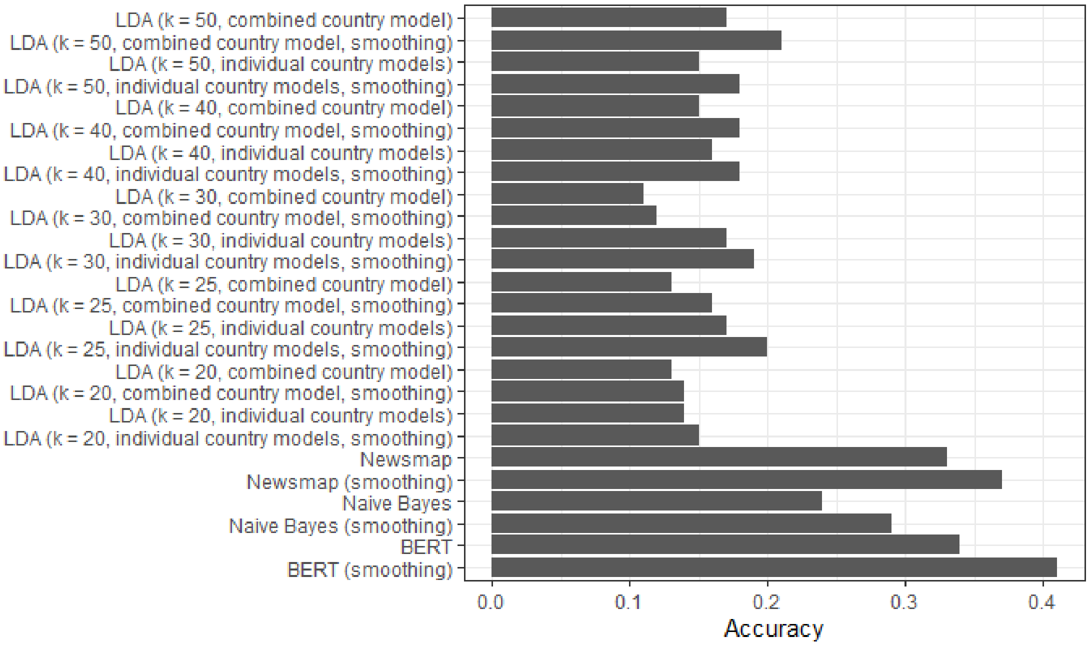
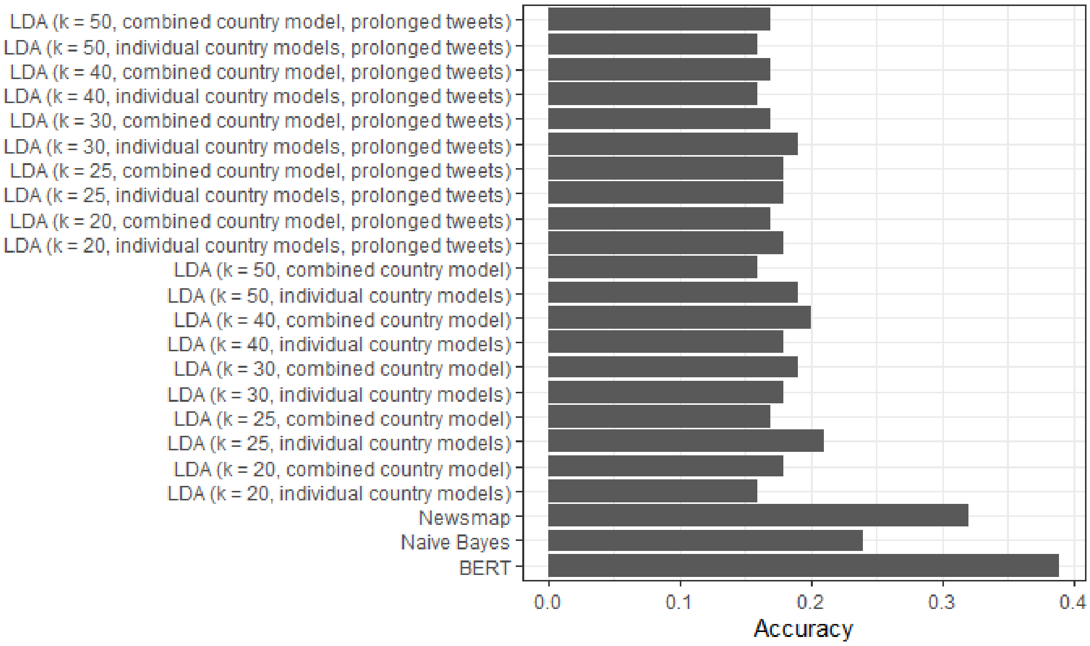


Figure A4 COMPARISON OF ALL MODEL PERFORMANCES (ACCURACY) FOR TWEETS



Appendix E

Table A3 - Table A6 compare the model performances (accuracy) per text type and per country.

Table A3 COMPARISON OF MODEL ACCURACIES PER COUNTRY FOR MANIFESTO CLASSIFICATION

Type	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naïve Bayes	Naïve Bayes (Smoothing)	BERT	BERT (Smoothing)
Austria	0.18	0.23	0.35	0.43	0.41	0.46	0.53	0.59
Germany	0.19	0.23	0.36	0.40	0.45	0.44	0.57	0.54
Switzerland	0.19	0.25	0.36	0.43	0.46	0.50	0.54	0.59

Table A4 COMPARISON OF MODEL ACCURACIES PER COUNTRY FOR PRESS RELEASE CLASSIFICATION

Type	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naïve Bayes	Naïve Bayes (Smoothing)	BERT	BERT (Smoothing)
Austria	0.15	0.18	0.33	0.39	0.27	0.32	0.37	0.45
Germany	0.17	0.22	0.34	0.41	0.29	0.35	0.41	0.52
Switzerland	0.07	0.07	0.32	0.37	0.31	0.35	0.40	0.45

Table A5 COMPARISON OF MODEL ACCURACIES PER COUNTRY FOR PARLIAMENTARY SPEECH CLASSIFICATION

Type	LDA	LDA (Smoothing)	Newsmap	Newsmap (Smoothing)	Naïve Bayes	Naïve Bayes (Smoothing)	BERT	BERT (Smoothing)
Austria	0.18	0.20	0.34	0.37	0.19	0.20	0.32	0.36
Germany	0.16	0.20	0.32	0.37	0.25	0.32	0.33	0.40
Switzerland	0.09	0.10	0.32	0.39	0.29	0.37	0.40	0.50

Table A6 COMPARISON OF MODEL ACCURACIES PER COUNTRY FOR TWEET CLASSIFICATION

Type	LDA	Newsmap	Naïve Bayes	BERT
Austria	0.21	0.30	0.22	0.37
Germany	0.21	0.31	0.22	0.38
Switzerland	0.22	0.36	0.31	0.44