Deep Learning for News Topic Identification in Limited Supervision and Unsupervised Settings

Arezoo Hatefi
Today is neither the beginning nor the end of the world.

Oh, what sorrow and joy lie hidden behind the curtain!

If you are on the path, do not grieve over the distance or delay.

You know that reaching it takes the step of time.

Saaye (Shadow)
Abstract

In today’s world, following news is crucial for decision-making and staying informed. With the growing volume of daily news, automated processing is essential for timely insights and in aiding individuals and corporations in navigating the complexities of the information society. Another use of automated processing is contextual advertising, which addresses privacy concerns associated with cookie-based advertising by placing ads solely based on web page content, without tracking users or their online behavior. Therefore, accurately determining and categorizing page content is crucial for effective ad placements. The news media, heavily reliant on advertising to sustain operations, represent a substantial market for contextual advertising strategies.

Inspired by these practical applications and the advancements in deep learning over the past decade, this thesis mainly focuses on using deep learning for categorizing news articles into topics of varying granularity. Considering the dynamic nature of these applications and the limited availability of relevant labeled datasets for training models, the thesis emphasizes developing methods that can be trained effectively using unlabeled or partially labeled data. It proposes semi-supervised text classification models for categorizing datasets into predefined coarse-grained topics, where only a few labeled examples exist for each topic, while the majority of the dataset remains unlabeled. Furthermore, to better explore coarse-grained topics within news archives and streams and overcome the limitations of predefined topics in text classification the thesis suggests deep clustering approaches that can be trained in unsupervised settings.

Moreover, to address the identification of fine-grained topics, the thesis introduces a novel story discovery model for monitoring event-based topics in multi-source news streams. Given that online news reporting often incorporates diverse modalities like text, images, video, and audio to convey information, the thesis finally initiates an investigation into the synergy between textual and visual elements in news article analysis. To achieve this objective, a text-image dataset was annotated, and a baseline was established for event-topic discovery in multimodal news streams. While primarily intended for news monitoring and contextual advertising, the proposed models can, more generally, be regarded as novel approaches in semi-supervised text classification, deep clustering, and news story discovery. Comparison with state-of-the-art baseline models
demonstrates their effectiveness in addressing the respective objectives.
Sammanfattning


Inspirerad av dessa praktiska tillämpningar och det senaste decenniets framsteg inom djupinlärning fokuserar denna avhandling främst på användandet av djupinlärning för att ämneskategorisera nyhetsartiklar på varierande nivåer av granularitet. Med avseende på hur dynamiska dessa applikationer är och den begränsade tillgängligheten av relevant annoterad data att träna på, betonar avhandlingen utvecklingen av metoder som effektivt kan tränas med oannoterad eller delvis annoterad data. Avhandlingen föreslår semi-övervakade textklassificeringsmodeller för att kategorisera datamängder i fördefinierade ämnen på hög nivå, där endast ett fåtal annoterade exempel finns för varje ämne, medan största delen av datamängden saknar annotering. För att bättre utforska ämnen lämpliga för nyhetsarkiv och strömmad data, samt adressera begränsningarna som fördefinierade ämnen för textklassificering medför, föreslås djupa klustringsmetoder som kan tränas i oövervakade miljöer.

Utöver detta, och för att förbättra identifieringen av detaljerade ämnen, introducerar avhandlingen en ny modell för upptäckt av berättelser för att övervaka händelsebaserade ämnen i nyhetsströmmar med flera källor. Med tanke på att onlinerapportering av nyheter ofta använder sig av en kombination av olika modaliteter som text, bilder, video och ljud för att förmedla information, undersöker avhandlingen också samverkan mellan textuella och visuella element i analys av nyhetsartiklar. För att uppnå detta mål annoterades en text-bild-datamängd, och en referensmodell för upptäckt av händelsesrelaterade ämnen i multimodala nyhetsströmmar utvecklades. Även om de i första hand är
avsedda för nyhetsövervakning och kontextuell reklam, kan de föreslagna modellerna merallmänt betraktas som nya tillvägagångssätt för semi-övervakad textklassificering, djup klustring och upptäckt av nyheter. En jämförelse med state-of-the-art visar modellernas effektivitet.
Preface

This thesis contains the following papers.


Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Frank Drewes and Johanna Björklund for their support, guidance and trust throughout the journey of my PhD. Thanks for believing in me and helping me grow as a researcher and also as an individual. Frank, I deeply appreciate all the insightful discussions and valuable feedback you provided me with. Johanna, you inspired me with your knowledge, experience, management, and personality. I would also like to extend my heartfelt thanks to my co-advisor, Xuan-Son Vu, for his guidance and assistance in finding my research direction when I was at a crossroads. Additionally, I am grateful to my reference person, Patrik Eklund, as well as Monowar Bhuyan for their valuable guidance and contributions.

This thesis was made possible through the funding and support provided by the Industrial Doctoral School at Umeå University and Aeterna Labs. Their contributions have been instrumental in the successful completion of this thesis, and I am grateful for the opportunities they have provided me with. I want to extend special thanks to the amazing team at Aeterna Labs, particularly Mona and Anton. Thanks for the great collaboration and friendly environment you provided me with to peruse my ideas and do my research.

Next, I want to express my gratitude to all my colleagues and friends in the Foundations of Language Processing group at Umeå University for their support and the memorable moments we shared during launches, fikas, and other enjoyable events. Special thanks to Anna and Adam for their assistance in translating the abstract of the thesis into Swedish. I am also thankful to all my colleagues and friends at the Department of Computing Science.

I am deeply indebted to my parents, as well as my extended family, including my in-laws, for all the love and support during all these years. I am immensely grateful to my parents for their unwavering support throughout my entire life, especially during my PhD journey, where their emotional support was a constant source of strength. They have always been there for me, and I deeply appreciate their presence and guidance. To my dear sister and brother, Atefe and Reza, I am incredibly thankful to have you in my life.

Lastly, to my beloved Abbas for endless support and love and for standing by me through the emotional ups and downs of my PhD journey. My life is worth nothing without you. I Love you dearly.
To my beloved Abbas,
and to my parents, Iran and Ali,
and to all the courageous Iranian women #Mahsa-Amini.
# Abbreviations

Table 1: List of terminologies and abbreviations used in the thesis

<table>
<thead>
<tr>
<th>#</th>
<th>Abbreviation</th>
<th>Full form</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NLP</td>
<td>Natural Language Processing</td>
</tr>
<tr>
<td>2</td>
<td>IR</td>
<td>Information Retrieval</td>
</tr>
<tr>
<td>3</td>
<td>AI</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>4</td>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>5</td>
<td>MLP</td>
<td>Multilayer Perceptron</td>
</tr>
<tr>
<td>6</td>
<td>CNN</td>
<td>Convolutional Neural Network</td>
</tr>
<tr>
<td>7</td>
<td>RNN</td>
<td>Recurrent Neural Network</td>
</tr>
<tr>
<td>8</td>
<td>MLM</td>
<td>Masked Language Modeling</td>
</tr>
<tr>
<td>9</td>
<td>LM</td>
<td>Language Model</td>
</tr>
<tr>
<td>10</td>
<td>PLM</td>
<td>Pre-trained Language Model</td>
</tr>
<tr>
<td>11</td>
<td>LLM</td>
<td>Large Language Model</td>
</tr>
<tr>
<td>12</td>
<td>BERT</td>
<td>Bidirectional Encoder Representations from Transformers [28]</td>
</tr>
<tr>
<td>13</td>
<td>MPL</td>
<td>Meta Pseudo Labels [115]</td>
</tr>
<tr>
<td>14</td>
<td>TF-IDF</td>
<td>Term Frequency-Inverse Document Frequency</td>
</tr>
<tr>
<td>15</td>
<td>UDA</td>
<td>Unsupervised Data Augmentation [171]</td>
</tr>
<tr>
<td>16</td>
<td>PCA</td>
<td>Principal Components Analysis [165]</td>
</tr>
<tr>
<td>17</td>
<td>UMAP</td>
<td>Uniform Manifold Approximation and Projection [97]</td>
</tr>
<tr>
<td>18</td>
<td>LDA</td>
<td>Latent Dirichlet Allocation [13]</td>
</tr>
<tr>
<td>19</td>
<td>DEC</td>
<td>Deep Embedded Clustering [169]</td>
</tr>
<tr>
<td>20</td>
<td>InfoNCE</td>
<td>Normalized Cross-Entropy with Information Maximization [108]</td>
</tr>
<tr>
<td>21</td>
<td>TDT</td>
<td>Topic Detection and Tracking Allan [1]</td>
</tr>
<tr>
<td>22</td>
<td>CLIP</td>
<td>Contrastive Language-Image Pre-Training [123]</td>
</tr>
</tbody>
</table>
# Contents

1 Introduction 1
   1.1 Automatic News Analysis ............................. 1
   1.2 Contextual Advertising ............................... 2
   1.3 Research Problem and Questions ..................... 3
   1.4 Thesis Outline ...................................... 5

2 Deep Learning 7
   2.1 A Brief Introduction to Deep Learning ............... 7
   2.2 Learning Paradigms .................................. 9
   2.3 Data Dependency in Deep Learning .................. 11
      2.3.1 Transfer Learning ............................... 11
      2.3.2 Data Augmentation .............................. 14
      2.3.3 Pseudo-labeling and Teacher-Student Architecture 15

3 Coarse-Grained News Topic Identification 17
   3.1 News Classification .................................. 17
      3.1.1 Classification .................................. 18
      3.1.2 Text Classification ............................... 20
      3.1.3 Semi-Supervised Text Classification .............. 21
   3.2 News Clustering ...................................... 22
      3.2.1 Clustering ....................................... 22
      3.2.2 Traditional Clustering Algorithms ............... 22
      3.2.3 Dimension Reduction .............................. 23
      3.2.4 Deep Clustering .................................. 24

4 Event-based Topic Discovery in News Streams 29
   4.1 Topic Detection and Tracking ......................... 29
   4.2 Online Clustering .................................... 32
   4.3 Multimodal News Streams .............................. 34
      4.3.1 Deep Learning for Multimodal Data ................ 35
5 Summary of Contributions

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Paper I</td>
<td>38</td>
</tr>
<tr>
<td>5.2</td>
<td>Paper II</td>
<td>39</td>
</tr>
<tr>
<td>5.3</td>
<td>Paper III</td>
<td>41</td>
</tr>
<tr>
<td>5.4</td>
<td>Paper IV</td>
<td>42</td>
</tr>
<tr>
<td>5.5</td>
<td>Paper V</td>
<td>43</td>
</tr>
</tbody>
</table>

Paper I 63

Paper II 76

Paper III 118

Paper IV 139

Paper V 159
Chapter 1

Introduction

This chapter motivates the development of systems capable of effectively processing and interpreting the wealth of information provided by news sources. It introduces contextual advertising as the real-world application that served as the motivation behind this thesis. Furthermore, it highlights the objectives of the thesis and concludes with an overview of the thesis structure.

1.1 Automatic News Analysis

In today’s world, news serves as a cornerstone for information, awareness, and societal cohesion. It shapes public opinions, guides decision-making, and reflects the evolving global landscape. News monitoring is crucial for individuals, aiding informed decision-making and crisis awareness. Similarly, it is valuable for companies, providing insights into competitors, industry trends, and market developments, supporting strategic planning, positioning, risk management, and market intelligence.

A substantial and continually increasing volume of daily news articles, exemplified by Reuters’ production of approximately 5,000 articles from 2,500 journalists\(^1\), underscores the information flow. Additionally, social media has emerged as a significant news channel [58], especially during crises, influencing public discourse. In this extensive news landscape, automated processing becomes crucial, swiftly navigating through vast content to keep individuals and companies updated on evolving events. Also, by leveraging different sources, automated news processing can offer a broader range of perspectives, giving a well-rounded understanding of events.

In Natural Language Processing (NLP) and Information Retrieval (IR), numerous research lines are dedicated to the study of news, including:

- **News Topic Modeling**: techniques for automatically identifying and

\(^1\)https://www.reutersagency.com/en/about/about-us/
categorizing topics within news articles to understand and organize the content effectively [86, 69, 161, 65, 118].

- **Topic Detection and Tracking**: methods for automatically identifying, monitoring, and organizing emerging topics or events from a continuous flow of news [1, 72, 102, 137, 182, 34, 59].

- **Sentiment Analysis**: methods to determine sentiment and opinions expressed in news articles, helping to understand public reactions and attitudes towards different topics [103, 27, 6, 178].

- **Fake News Detection**: methods to identify misinformation and fake news, including the development of algorithms that assess the credibility and reliability of news sources [168, 195, 194, 179, 193, 67, 98].

- **Personalized News Recommendation**: algorithms for personalized news recommendations based on individual preferences, browsing history, and user behavior [106, 167, 3, 56, 116].

- **Summarization**: techniques to automatically generate concise and informative summaries of news articles, focusing on both extractive and abstractive summarization [11, 146, 174, 190].

- **Multimodal Analysis in News**: research exploring the integration of text with other modalities (images, videos) in news analysis to provide a more comprehensive understanding of news content [22, 65, 161, 105, 117].

The central focus of this doctoral thesis is on the first and second research domains, specifically the identification of topics within news articles at different levels of granularity. This endeavor encompasses the classification of news into broader categories such as “sports” and “politics,” as well as more detailed event-based topics like “a plane crash in Malaysia.” The research includes analyzing both static collections of news articles and continuous streams of news content. The motivation behind this research stems from the application of contextual advertising. Furthermore, there exists an interest in multimodal news processing, exploring the correlation between news text and images for the purpose of news topic identification.

### 1.2 Contextual Advertising

Traditional automated advertising relies on cookies and users’ browsing and shopping histories. However, growing privacy concerns have prompted advertisers to explore alternative approaches. Contextual advertising, as a privacy-friendly and less invasive alternative to cookie-based advertising, has emerged in response to these concerns. Contextual advertising, also known as cookie-less advertising, involves placing ads on web pages based solely on their content,
without tracking users and their online behavior. For instance, this could mean displaying ads for an online Artificial Intelligence (AI) course on a news article about AI. Figure 1.1 illustrates a contrast between conventional and contextual advertising methods.

Despite the advantages of cookie-based behavioral advertising, which enables deeper personalization and utilizes browsing history as a strong indicator of buying readiness, many companies are now shifting their advertising approaches toward contextual advertising. An essential advantage is that companies can avoid dealing with the constantly evolving regulations, legislation, and shifting attitudes toward privacy associated with tactics that use cookies to track user online behavior. For instance, General Data Protection Regulation (GDPR) [128] has introduced strict regulations on user consent and data privacy, significantly affecting the use of cookies for advertising purposes. This makes it increasingly challenging for companies to rely solely on cookie-based advertising. The process of obtaining valid consent can be complex, and user rejection of cookies due to privacy concerns is on the rise. Moreover, companies have concerns about brand safety and reputation. Contextual advertising allows them to select the specific contexts in which they want their ads to appear or not appear, providing greater control over the environments associated with their brand. Additionally, contextual advertising prioritizes current context over past behavior, ensuring that personalized ads align with users’ immediate interests and needs. By focusing on immediate relevance, it offers a certain level of appropriateness without compromising privacy.

Contextual advertising identifies relevant content for ads by aligning the advertising campaign’s keywords or topics with the central theme of the webpages. In this thesis, our focus is on news websites, presenting different approaches for news topic identification. It is important to note that while our motivation stems from an industrial application, the proposed methods can be viewed as general algorithms for document topic identification.

1.3 Research Problem and Questions

As outlined in earlier sections, the primary objective of this thesis is to categorize news articles into distinct topics, spanning various levels of granularity, for subsequent application in contextual programmatic advertising. News articles typically convey their messages through a multimodal approach, utilizing diverse modalities such as text, image, and video. Understanding the interplay between these modalities and understanding their respective contributions to topic identification is also an aspect studied in this thesis.

The accessibility of data intended for topic identification varies across different situations. In some cases, all data is readily accessible, while in others, the setting is more dynamic, presenting data as a continuous stream gradually accessible to the model. The dynamic scenario introduces the potential for various topic behaviors, such as emergence, disappearance, distribution shift,
Figure 1.1: A conceptual comparison between cookie-based advertising and contextual advertising within a programmatic advertising framework. The former relies on personal information, while the latter solely utilizes news content.
splitting, and merging, thereby complicating the task at hand. In such dynamic situations, the model must dynamically adapt to changes in the data stream to effectively predict future topics.

Deep learning, a specialized area of machine learning, is primarily focused on representation learning and has shown remarkable success across various domains in recent years. Its accomplishments include achieving state-of-the-art results, often leveraging the computational power of GPUs and TPUs to train deep neural networks on large datasets. Motivated by the effectiveness of deep learning, this thesis seeks to investigate its use for topic identification, especially in situations where labeled data is limited or unavailable.

Consequently, the thesis addresses four pivotal research questions:

**RQ1** How can news topics be automatically identified across various granularity levels?

**RQ2** What effective methodologies can be employed to integrate deep learning into the investigation of news topics when labeled data is scarce or unavailable?

**RQ3** How can deep learning techniques be utilized for topic identification in news streams while effectively addressing challenges associated with changes in topic focus and evolution over time?

**RQ4** What is the interrelation between different modalities within multimodal news, and how can these modalities be harnessed for the purpose of topic identification?

### 1.4 Thesis Outline

The thesis is structured into four main chapters. Chapter 2 provides a brief introduction to deep learning and learning paradigms. Moreover, it discusses the data-dependency of deep learning methods as a challenge in the field and presents some solutions, such as transfer learning, data augmentation, and pseudo-labeling, to address this challenge. These techniques have been utilized in various parts of the thesis for training models in semi-supervised and unsupervised settings. Chapter 3 is centered on coarse-grained topic identification for news articles. It introduces classification and clustering tasks, along with other relevant preliminaries essential for comprehending Papers I, II, and III, which focus on addressing research questions RQ1 and RQ2 concerning coarse-grained topics. Paper I and Paper II introduce semi-supervised classification models utilizing deep learning for coarse-grained news topic identification, while Paper III suggests deep clustering in unsupervised settings for the same purpose. Chapter 4, is focused on identification of fine-grained event-based topics in streams of news articles and provides the background for Papers IV and V, which tackle research questions RQ3 and RQ4, respectively. Chapter 5
concludes the thesis by summarizing the research contributions. Furthermore, the thesis includes five papers related to the research.
Chapter 2

Deep Learning

2.1 A Brief Introduction to Deep Learning

Machine Learning (ML), encompasses algorithms designed to learn from data presented as feature vectors and predict outcomes for new data. These feature vectors encompass features denoting distinct aspects or attributes of the data, carefully crafted by human experts through a process known as feature engineering. For example, when predicting the appropriate food category for recipes, a comprehensive feature set might encompass a range of ingredients and cooking techniques such as grilling, baking, frying, steaming, and beyond.

Deep Learning, a subfield of Machine Learning, specializes in data representation learning, automating the feature extraction process by eliminating the need for human intervention. Deep learning algorithms ingest unstructured raw data, such as text and images, and autonomously infer the crucial features for decision making [76]. The cornerstone of deep learning lies in deep neural networks, which are composed of multiple interconnected layers of artificial modules many of which compute non-linear input–output mappings.

Deep Feedforward Networks, also known as Multilayer Perceptrons (MLPs), serve as fundamental modules in deep learning, characterized by multiple layers of neurons comprising an input layer, one or more hidden layers, and an output layer. The addition of extra hidden layers enhances the model’s predictive capability, particularly with abundant training data. In an MLP, each neuron in a layer connects to every neuron in the subsequent layer, forming a fully connected network structure. Transitioning from one layer to the next, each neuron calculates a weighted sum of its inputs from the preceding layer and passes the result through a non-linear function such as the Rectified Linear Unit (ReLU), defined as the half-wave rectifier \( f(z) = \max(z, 0) \). When a neuron’s output surpasses its threshold, it becomes activated, transmitting data to the subsequent layer. Conversely, if the output falls below the threshold, no data is transmitted. This sequential computation process across the network is termed forward propagation.
During the training process, the learning algorithm fine-tunes the weights of the network using a method known as backpropagation [134, 132]. This technique, rooted in the chain rule for derivatives [148], computes the gradient of the training objective function with respect to the network’s weights. Essentially, it quantifies how each weight contributes to the overall prediction error. Backpropagation operates by recursively applying the chain rule to propagate gradients backward through the network layers, starting from the output and moving towards the input. These gradients serve as crucial information for optimization algorithms like variants of gradient descent [133] to adjust the network weights. Through this iterative refinement process, the network steadily improves its predictive performance.

MLPs have been widely used in various fields, including image recognition, natural language processing, and financial forecasting. Despite their simplicity compared to more complex architectures, MLPs remain powerful tools in machine learning and serve as the foundation for many deep learning models.

In addition to MLPs as the most basic form of deep neural networks, deep learning encompasses diverse neural network architectures tailored to tackle specific challenges or datasets.

Convolutional Neural Networks (CNNs) are a type of deep learning models primarily designed for processing and analyzing visual data, such as images and videos [76]. They consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers use filters to extract features from input images, while pooling layers reduce spatial dimensions. CNNs leverage parameter sharing and hierarchical feature learning to efficiently extract meaningful patterns from the data, making them highly effective for tasks such as image classification, object detection, and image segmentation.

Recurrent Neural Networks (RNNs) are a type of neural network designed to work with sequential data, where the order of the data points matters [134]. Unlike feedforward neural networks, which process each input independently, RNNs maintain a memory of previous inputs by using loops within the network architecture. This memory enables RNNs to capture temporal dependencies and patterns in sequential data, making them well-suited for tasks such as time series prediction, natural language processing, speech recognition, and handwriting recognition. RNNs are characterized by their ability to handle inputs of varying lengths and their capability to learn from past information to make predictions about future data points. However, traditional RNNs suffer from the vanishing gradient problem, which can hinder their ability to learn long-range dependencies. To address this issue, variants of RNNs such as Long Short-Term Memory (LSTM) networks [52] and Gated Recurrent Units (GRUs) [24] have been developed, which incorporate mechanisms to better retain and update information over long sequences.

Autoencoders aim to learn a compressed representation of input data by first encoding it into a lower-dimensional space (encoder) and then reconstructing it from this representation (decoder) [48]. Autoencoders are commonly used
for tasks like data denoising, dimensionality reduction, and anomaly detection.

The Transformer architecture, introduced in [162], is a groundbreaking model for natural language processing. It has an encoder-decoder architecture. Initially, the encoder processes the input sequence, converting it into a set of contextualized representations. Subsequently, the decoder utilizes these representations to produce the output sequence, attending to relevant sections of the input sequence through cross-attention mechanism. While the encoder and decoder are often used together in sequence-to-sequence tasks like machine translation, they can also be used separately for specific tasks that only require encoding or decoding functionality. For instance, encoder models can be utilized separately for tasks like text classification or named entity recognition. Similarly, decoder models can be employed independently for tasks such as text generation or language modeling. The Transformer utilizes self-attention to capture long-range dependencies between words in a sentence efficiently. Self-attention empowers the network to consider every other word and determine the significance it should assign to different words when generating representation of a word in the input sequence. Other key components are multi-head attention for focusing on different aspects of the input, positional encodings to provide sequential order information, feedforward neural networks for complex feature extraction, and layer normalization with residual connections for stable training. Unlike recurrent models, Transformers process the entire input sequence in parallel, making them highly efficient for both training and inference. Transformers have revolutionized NLP by outperforming older models like RNNs and CNNs, making them widely adopted in both research and industry for various tasks such as machine translation and text generation.

2.2 Learning Paradigms

The supervised, unsupervised, semi-supervised, and reinforcement learning paradigms are different approaches to training machine learning models, each with its own characteristics and applications.

- **Supervised Learning** involves training a model on a dataset consisting of input-output pairs, where the input data is associated with corresponding labels or target values. The goal is to learn a mapping from inputs to outputs based on the provided examples, enabling the model to make predictions on new, unseen data [12]. Common tasks in supervised learning include classification (predicting discrete labels) and regression (predicting continuous values). Examples of supervised learning algorithms include decision trees [119], support vector machines [26], and neural networks [47].

- **Unsupervised Learning** involves training the model on input data without explicit labels or target values. Instead, the model learns patterns, hidden structures, or representations inherent in the data without
guidance [45]. Clustering algorithms, dimensionality reduction techniques like Principal Component Analysis (PCA) [63], and generative models like Gaussian mixture models [130] are common examples of unsupervised learning algorithms. Self-supervised learning is a special case of unsupervised learning that has gained popularity in deep learning as a way to leverage large amounts of unlabeled data for representation learning and for pre-training models which can then be fine-tuned on smaller labeled datasets for specific tasks. In self-supervised learning, a model is trained using supervision signals that are automatically generated from the input data itself, without requiring manually labeled data. This typically involves creating auxiliary tasks or objectives that are related to the main task of interest but do not require explicit human annotation. Examples of deep networks trained in a self-supervised manner include Autoencoders [49], generative models such as Generative Adversarial Networks (GANs) [39] and Variational Autoencoders (VAEs) [68], as well as Pre-trained Language Models (PLMs) [28].

- **Semi-Supervised Learning** combines elements of both supervised and unsupervised learning paradigms [21]. In this approach, the model is trained on a dataset that contains a small amount of labeled data along with a larger amount of unlabeled data. Semi-supervised learning is particularly useful when obtaining labeled data is expensive or time-consuming, as it allows using abundant unlabeled data to improve model performance. Unlabeled data can contribute to the learning process in several ways. It can be used for regularization purposes, particularly in consistency training [171]. Consistency regularization techniques encourage the model to produce consistent predictions for similar examples in the unlabeled data. By penalizing inconsistencies between predictions on different perturbations of the same input, the model learns to generalize better and become more robust. Unlabeled data can also be used as a form of pseudo-labeled data [78, 115]. In this approach, the model generates predictions for the unlabeled data, treating these predictions as pseudo-labels. The model is then trained on a combined dataset consisting of both labeled and pseudo-labeled data.

- **Reinforcement Learning (RL)** is a learning paradigm which is commonly used in scenarios where explicit supervision is not available, and the agent must learn through trial and error. The agent learns by interacting with the environment, receiving feedback in the form of rewards or penalties, and adjusting its actions accordingly to achieve its goals. Reinforcement learning has applications in various domains, including robotics, game playing, autonomous driving, and recommendation systems. This line of algorithms are not the focus of this thesis. However, interested readers are recommended to see Sutton and Barto [151] for detailed information.
2.3 Data Dependency in Deep Learning

The data dependency challenge in deep learning refers to the reliance of deep neural networks on large amounts of labeled data for effective training. Deep learning models often require massive datasets to learn complex patterns and achieve high performance on various tasks. However, collecting labeled data can be costly, time-consuming, and sometimes impractical. Over the years, researchers have devised numerous methods to address this challenge. Here, we explain some of these techniques that are relevant to the field of NLP including transfer learning, data augmentation, and pseudo labeling.

2.3.1 Transfer Learning

In transfer learning, knowledge gained from solving one problem is applied to a different but related problem. In the context of deep learning, transfer learning involves taking a model trained on one task and fine-tuning it on a different task. This allows the model to leverage knowledge learned from the first task to improve its performance on the second task, especially when the second task has limited training data. In the context of natural language processing, word embeddings such as Word2Vec [101], GloVe [110], and FasText [15], where words are represented as dense vectors, were used to transfer knowledge from large text corpora to downstream NLP tasks however, pre-trained language models like BERT [28] have popularized and significantly advanced the application of transfer learning in NLP.

Pre-trained Language Models (PLMs) serve as a foundation for transfer learning in NLP. The key idea is acquiring a general, latent representation of language through a generic task, then using this knowledge for various NLP tasks. Language modeling, where the model predicts a word based on its context, serves as one such generic task due to the abundance of self-supervised text available for training. The process of training a deep neural network with a language modeling objective on a large corpus is termed pre-training. However, to effectively utilize the pre-trained model for downstream NLP tasks, further training or task adaptation is typically required. Existing task adaptation methods include fine-tuning the PLMs for the specific task, prompting the PLMs to execute the desired task, or reformulating the task as a text generation problem.

Almost all widely-used PLMs, such as those from the GPT series [16, 120, 121], BERT [28] and its variants, BART [79], and T5 [125], are built upon the Transformer architecture [162]. A Transformer-based language model can fall into one of three architectures: decoder-only (e.g., GPT [120] and Gopher [124]), encoder-only (e.g., BERT [28] and XLM-R [25]), or encoder-decoder (e.g., BART [79], T5 [125], and T0 [136]). Furthermore, models can be trained using different objectives: autoregressive training (predicting the next word based on preceding context), masked language modeling (MLM) (filling in the missing word, i.e., predicting the masked word given surrounding context), or
various denoising tasks where the model must undo some form of corruption in the original sequence, such as sentence permutation, token deletion, or span deletion. Typically, though not necessarily, decoder-only models are trained with an autoregressive objective, encoder-only models utilize MLM for training, and encoder-decoder architectures are trained on denoising tasks or MLM.

While autoregressive models process input sequentially, masked language models predict a masked word based on all other words in the sequence offering greater contextual information. During MLM training, a random subset of tokens in the input text sequence is masked using a special token [MASK], and the model is trained to predict these masked tokens based on both left and right contexts. Hence, the training goal is to optimize the log-likelihood:

$$\sum_i m_i \log(P(x_i|x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n); \theta_T)$$

where $m_i \in \{0, 1\}$ indicates whether $x_i$ is masked or not, and $\theta_T$ denotes the model parameters. MLMs incorporate multiple transformer encoder layers [162] to progressively acquire meaningful representations. Prominent examples include BERT [28], RoBERTa [90], and XLM-R [25].

Lately, researchers have tried to enhance the adaptability of PLMs to specific tasks by further pre-training them with task-specific masking techniques. This method, referred to as objective masking, seeks to incorporate downstream task-related information into general PLMs through masking [64, 188, 155, 145, 41]. Gu et al. [41] proposed a three-stage framework for text classification by adding a task-guided pre-training stage with selective masking between general pre-training and fine-tuning stages. The method first finds the important words for the downstream task in the in-domain unsupervised dataset using a binary classifier trained on the supervised task-specific dataset (which has already been annotated automatically with word importance information) and then masks them in the task-guided pre-training stage. In Paper II, we utilize objective masking to incorporate topical information, collected in an unsupervised way based on statistical information from the unlabeled data, into the PLM for topic classification.

Fine-tuning adjusts the contextual representations obtained during pre-training for distinct NLP tasks. Typically, for classification tasks such as sentiment analysis, natural language inference, and semantic similarity, one or two feed-forward classification layers, referred to as prediction heads [166], are appended on top of the PLM. The classification head transforms the contextualized embeddings generated by the language model into predictions for desired classes. Both the output layers and the PLM undergo training simultaneously in an end-to-end setup, with the major computational load allocated to fine-tuning the LM. It is crucial to carefully select the learning rate for the weights of the feed-forward layer(s) and for the PLM in this configuration. Given that the PLM is already extensively trained, a low learning rate is advisable, especially for smaller datasets. Conversely, the feed-forward layer weights, which are initialized randomly, necessitate considerable training. The word embeddings are
derived either directly from the top layer of the language model or through a concatenation or weighted average of the top $n$ (typically $n = 4$) layers [113]. The text representation can then be computed by taking a weighted average of the word embeddings or the representation of the special [CLS] token. It is worth noting that certain tasks, such as parsing tasks, demand significant additional architecture atop a PLM [183]. In such instances, ample training data and computational resources are essential to train both the task-specific architecture and effectively fine-tune the PLM.

**Prompting** is the practice of inserting natural language text or continuous vectors in the input to guide PLMs in executing particular tasks. The objective of prompting is to simplify the downstream task for the language model by utilizing the prompts as contextual cues. Prompting serves as a knowledge probing technique for PLMs, enabling evaluation of their acquired knowledge for specific tasks [114]. Two common approaches to prompt learning are **template-based learning** and **in-context learning**.

**Template-based learning** reformulates NLP tasks into tasks resembling pre-training tasks of language models, such as MLM, by employing templates. This strategy effectively utilizes the knowledge learned during pre-training, resulting in a significant reduction in the number of task-specific training examples needed, which is particularly advantageous in scenarios with limited data [74]. Le Scao and Rush [74] conducted an extensive analysis to quantify the advantages of prompts in classification tasks. Their study involved controlled fine-tuning across various tasks and data sizes, demonstrating that the use of prompts consistently enhances performance compared to relying solely on traditional fine-tuning methods. For supervised template-based prompt learning, labeled examples are converted into “natural” text using carefully crafted templates with open slots. Subsequently, solving the tasks becomes a matter of filling these slots with words or phrases using PLMs and then mapping these outputs to task-specific labels via a verbalizer. **Cloze-style** templates introduced in [138] are one of the widely used templates. Table 2.1 shows examples of this approach, for text classification, sentiment classification, textual entailment, and probing for facts.

<table>
<thead>
<tr>
<th>Task</th>
<th>Cloze-style template</th>
<th>PLM’s output</th>
<th>Task-specific class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic classification</td>
<td>— News: [Article Text]</td>
<td>Politics</td>
<td>1</td>
</tr>
<tr>
<td>Sentiment classification</td>
<td>[Movie Review]. Overall, it was —.</td>
<td>Disappointing</td>
<td>Negative</td>
</tr>
<tr>
<td>Textual entailment</td>
<td>The cat is on the table? —, the cat is</td>
<td>No</td>
<td>FALSE</td>
</tr>
<tr>
<td></td>
<td>under the table.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probing for facts</td>
<td>Obama was the president of the —.</td>
<td>U.S.</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2.1: Examples of close-style prompting for different NLP tasks.

In **Paper IV** we used cloze-style prompting to make topic-aware document representation for topic identification.

**In-context learning** or learning from instructions and demonstrations is particularly efficient when applied to large generative PLMs, known as Large
Language Models (LLMs). These models were initially introduced in [121], featuring tens to hundreds of billions of parameters. They have demonstrated significant performance across various NLP tasks in zero-shot and few-shot settings. LLMs such as GPT-3 [16] exhibit the capability to handle diverse NLP tasks in a few-shot setting through in-context learning. In-context learning provides LLMs with instructions and a few input-output examples for a specific task, allowing them to produce desired outputs for new inputs without the need for gradient updates. In contrast to the easy implementation, there are certain limitations to consider for this prompting approach. Firstly, its success in few-shot tasks heavily relies on the sheer size of LLMs, limiting its applicability. Moreover, insights from [112] indicate that the few-shot performance of PLMs is very sensitive to the choice of prompts which limits the robustness of the approach.

2.3.2 Data Augmentation

Data augmentation was first introduced in computer vision to enhance the quantity and diversity of training examples by modifying the original data while maintaining its semantic significance [75, 143]. This process is relatively straightforward in computer vision, where techniques such as cropping, rotating, flipping, or color jittering can be applied to generate new examples without changing the underlying subject matter [141].

However, in NLP, due to the discrete nature of language, augmenting text while maintaining its meaning poses greater challenges. Even minor alterations can significantly change the meaning of the text. Nonetheless, researchers have proposed several promising methods for text augmentation including token-level random perturbation operations such as random insertion, deletion, and swap [164], back translation (translating the text into another language and then back into the source language) [140], replacing words with synonyms [187], utilizing PLMs to substitute words based on their contextual surroundings [71], and guided generation using large-scale generative language models [89, 88]. Another example is TMix [23], inspired by MixUp in computer vision [186], which interpolates two or more text instances and their labels in their respective hidden space. In Papers I and II, we used PLMs to generate augmented text for semi-supervised text classification.

Data augmentation methods are employed to expand labeled datasets when training data is insufficient, as exemplified by Augmented SBERT [154]. Additionally, these techniques can introduce noise to data to enhance model robustness, particularly in consistency training. Xie et al. [171] proposed to replace the traditional noise injection methods by high quality data augmentation such as back translation of textual data.
2.3.3 Pseudo-labeling and Teacher-Student Architecture

To tackle the data dependency issue in deep learning methods, one effective approach is to train the deep neural network in a semi-supervised manner. Semi-supervised learning encompasses various techniques like self-training [163, 2] and temporal ensembling [73]. Among these methods, pseudo labeling using a teacher-student architecture stands out as a commonly utilized approach.

The teacher-student architecture was initially used for knowledge distillation from a large teacher to a light-weight student while maintaining comparable performance with the teacher [50, 152]. Recently, the teacher-student architecture has found widespread application in various types of knowledge learning objectives including knowledge expansion [170, 144], knowledge adaptation [96, 156], and multi-task learning [37, 176].

Knowledge expansion aims to train a student model with superior generalizability and performance compared to the teacher model by leveraging the vast amount of unlabeled data in a semi-supervised fashion. In this approach, the capacity of the student model is either the same as or larger than that of the teacher model. To achieve this goal through an offline approach, the teacher model is initially trained or fine-tuned using labeled data. Subsequently, it generates predictions, termed pseudo-labels, for the unlabeled dataset. Both the labeled and pseudo-labeled datasets are then utilized to train the student model. This process facilitates the transfer of knowledge from the teacher to the student, potentially leading to improved performance as the student benefits from pseudo-labeled data and the application of regularization techniques such as data augmentation [170].

Despite the simplicity and computational efficiency of the offline learning scheme, it has a drawback: if the pseudo-labels predicted by the teacher network are inaccurate, confirmation bias may arise, as the student network may reinforce existing inaccuracies [4]. Consequently, the student may not surpass the performance of the teacher significantly. To mitigate the confirmation bias problem, Pham et al. [115] proposed Meta Pseudo Labels (MPL), an iterative training approach for both the teacher and student networks, resulting in enhanced performance for both networks. In each iteration, the teacher receives feedback from the student in the form of the student’s performance on the gold-labeled data and adjusts itself accordingly to predict more accurate pseudo-labels in the subsequent iteration. In Papers I and II, we employed pseudo-labeling for text classification, and in Paper III, it was used for clustering.
Chapter 3

Coarse-Grained News Topic Identification

This chapter provides the background knowledge for Papers I, II, and III, which are focused on addressing research questions RQ1 and RQ2. Paper I and Paper II propose semi-supervised classification models using deep learning for news topic identification. This is particularly relevant when predefined coarse-grained topics are of interest and there is insufficient labeled data available to effectively train the classifier. Paper III advocates for deep clustering in scenarios where a set of predefined classes is absent, yet there is a desire to explore coarse-grained topics within the news dataset. This chapter provides an introduction to both classification and clustering tasks, as well as other relevant preliminaries.

3.1 News Classification

Classification plays a crucial role in various applications within the News domain, such as fake news detection, sentiment analysis, and topic identification. To categorize news articles into topics, standard taxonomies like Interactive Advertising Bureau (IAB) tags\(^1\) and IPTC media topics\(^2\) are commonly utilized, offering predefined classes organized into hierarchical structures. The IAB provides a standardized categorization system tailored for classifying news content, aiding advertisers and publishers in effectively categorizing news articles. This taxonomy covers a wide range of topics, from “politics” and “sports” to “entertainment” and “technology”, with each category further subdivided for enhanced granularity. By ensuring consistency and clarity in labeling and organization, the IAB taxonomy facilitates targeted advertising and efficient content monetization for publishers.

\(^1\)https://www.iab.com/guidelines/content-taxonomy/
\(^2\)https://iptc.org/standards/media-topics/
Utilizing deep learning for categorizing news articles into predefined classes, such as those found in one of the layers of the IAB taxonomy, typically requires a substantial labeled dataset, which is often unavailable or requires significant manual effort to compile. The extensive range of potential topics further complicates the task of gathering training documents to construct a supervised classification model. Consequently, semi-supervised approaches have gained popularity in this field.

Prior studies in contextual advertising have explored the creation of labeled datasets for training classifiers using class-specific keywords and knowledge bases. Jin, Wanvarie, and Le [62] proposed a method to model contextual targeting as a lightly-supervised one-class classification problem. Their algorithm takes unlabeled documents and labeled keywords for the target class $c$ as input, generating a classifier $M_c$ specifically designed to identify documents belonging to class $c$. However, the dynamic nature of contextual advertising poses challenges in preparing effective keywords for each class and maintaining one-class classifiers, especially for large-scale applications. In a related study, Jin, Kadam, and Wanvarie [61] automated the process of mapping categories in the IAB taxonomy to category nodes in the Wikipedia category graph. Through label propagation across the graph, they obtained a list of labeled Wikipedia documents for training purposes.

To the best of our knowledge, no previous study has explored the utilization of pre-trained language models and their implicit knowledge, alongside deep learning techniques, for categorizing news articles specifically for contextual advertising purposes.

### 3.1.1 Classification

Classification is one of the fundamental and challenging problems in machine learning, with applications in various fields such as natural language processing, computer vision, and speech recognition. It involves categorizing a set of data instances into predefined classes. In classification tasks, the model aims to learn patterns and relationships in the data that distinguish between different classes, enabling it to accurately classify unseen instances based on their features or attributes. This process involves training the model on labeled data, where each data instance is associated with a known class label, and then evaluating its performance on unseen data to assess its ability to generalize to new examples [12]. Common evaluation metrics for classification tasks include:

- **Accuracy**: the proportion of correctly classified instances out of the total number of instances. It is a simple and intuitive metric but can be misleading in the presence of imbalanced classes.

- **Precision**: The proportion of true positive predictions out of all positive predictions made by the model. It measures the accuracy of positive predictions and is useful when the cost of false positives is high.
• Recall (Sensitivity): the proportion of true positive predictions out of all actual positive instances in the dataset. It measures the ability of the model to identify all positive instances and is important when the cost of false negatives is high.

• F1 Score: the harmonic mean of precision and recall. It provides a balance between precision and recall and is particularly useful when there is an imbalance between the classes.

• Area Under the ROC Curve (AUC-ROC): the area under the receiver operating characteristic (ROC) curve, which plots the true positive rate against the false positive rate at various threshold settings. AUC-ROC summarizes the performance of a classifier across all possible threshold settings and is particularly useful for imbalanced datasets.

These metrics provide different perspectives on the performance of a classification model and are chosen based on the specific requirements and characteristics of the dataset and task at hand. In this thesis accuracy has been used to measure classification performance.

Classification plays a crucial role in numerous real-world applications, including sentiment analysis [51, 33, 184, 139], document classification [181, 60], fraud detection [30, 111, 35], stock market prediction [175, 38, 46], face recognition [99, 91], disease diagnosis [131, 66], and so much more, where accurately categorizing data instances into meaningful classes is essential for decision-making and problem-solving.

Classification problems are typically categorized into four distinct types: binary, multi-class, multi-label, and imbalanced [12]. Binary classification involves tasks with two class labels. Multi-class classification pertains to tasks with more than two class labels. Multi-label classification refers to tasks where each example may have two or more class labels predicted. Imbalanced classification addresses tasks where the distribution of examples across classes is uneven. The classification problems in this thesis belong to the category of multi-class classification.

There are many classification algorithms in traditional machine learning including: naive bayes, decision trees, Support Vector Machines (SVM) and k-Nearest Neighbors (kNN) [12]. Also many deep learning models such as CNNs [76], RNNs [134], transformer Models (e.g., BERT [28], GPT [120]), and Autoencoders [48] can be used for classification tasks. Classic algorithms are often simpler and more interpretable, while deep learning algorithms tend to be more complex and capable of learning intricate patterns from large-scale data. The choice of algorithm depends on factors such as the size and nature of the dataset, computational resources, and the specific requirements of the classification task. Given the success of deep learning over traditional classification algorithms in NLP tasks, this thesis employs deep learning methodologies for news classification.
3.1.2 Text Classification

Text classification poses a significant challenge because it requires an effective representation of text capable of distinguishing between various classes. Initially, text was represented using the Term Frequency-Inverse Document Frequency (TF-IDF) approach, treating it as a bag of words [135]. This method involves creating a vocabulary, after which each piece of text is represented by a vector showcasing the TF-IDF values of the vocabulary words. The TF-IDF of word $t$ in document $d$ is computed as a product of the term frequency $tf(t, d)$ and the inverse document frequency $idf(t, d)$. $tf(t, d)$ is the relative frequency of word $t$ in document $d$ and $idf(t, d)$ is a measure of how much the word is common or rare across all documents. Various methods exist for computing these statistics. This representation has been commonly used with classical machine learning algorithms for text classification. However, the TF-IDF representation has limitations, such as its inability to account for sequential word orders and contextual information. Additionally, these vectors often have high dimensionality, resulting in computationally expensive operations and the curse of dimensionality problem [10].

To address the limitations of TF-IDF representations, word embeddings were introduced. Traditional word embeddings are static representations of words in a continuous vector space, where each word is assigned a fixed vector regardless of its context. Examples of traditional word embedding models include Word2Vec [101], GloVe [110], and FastText [15]. These models are typically learned using unsupervised learning techniques on large text corpora, analyzing the co-occurrence patterns of words within a context window in the training data. The underlying concept is that words appearing in similar contexts are likely to have similar meanings and should thus be close to each other in the vector space. These representations have commonly been employed with deep neural architectures such as LSTMs [52] and Autoencoders [48] for text classification tasks.

Following the rapid improvement of deep learning in the last decade, NLP has witnessed a drastic improvement in word and text representations resulted from emergence of pretrained language models. PLMs offer highly effective general-purpose contextual word embeddings that can be fine-tuned for specific domains. Contextual word embeddings are word representations that capture the meaning and context of words based on their surrounding context in a sentence or document. Unlike traditional word embeddings, which assign a single fixed vector to each word regardless of context, contextual word embeddings generate dynamic embeddings that vary based on the context in which the word appears. This allows contextual word embeddings to capture nuances in meaning and polysemy, as well as syntactic and semantic relationships between words. Today, encoding text with these language models has become standard practice as the initial step in text classification.
3.1.3 Semi-Supervised Text Classification

This section reviews prior research on semi-supervised text classification, exploring methodologies that use unlabeled data to enhance classification performance. Most of these methods have been used as baselines in Paper I and Paper II.

Several recent semi-supervised learning methods leverage consistency training on extensive amounts of unlabeled data [73, 153]. These techniques regularize model predictions to remain unaffected by minor levels of noise. Xie et al. [171] explored the impact of noise injection in consistency training and introduced Unsupervised Data Augmentation (UDA) as an alternative approach, replacing traditional noise injection with high-quality data augmentation techniques such as back translation for textual data.

Chen, Yang, and Yang [23] introduced TMix, a text augmentation technique interpolating two texts within their semantic hidden space. TMix promotes linear behavior across the training dataset. Additionally, they presented MixText, a new semi-supervised learning approach for text classification leveraging TMix. MixText employs a BERT-based text encoder equipped with TMix, followed by a linear classifier. During training iterations, it initially predicts labels for unlabeled data using the current model and subsequently trains the model with pseudo labeled data using TMix augmentation.

FLiText, introduced by Liu et al. [85], is a lightweight model designed for scenarios where resources are limited. Initially, an insipier network, based on a transformer model, is trained using both labeled and unlabeled data. Following this, the insipier network is distilled into a smaller CNN-based model using output-based distillation, which relies on the insipier’s output, and feature-based distillation, utilizing the layer weights of the insipier. FLiText significantly enhanced inference speed while maintaining or surpassing the state-of-the-art performance of lightweight models.

Xu, Liu, and Abbasnejad [172] proposed a novel approach to leverage the matching capability inherent in pre-trained language models like BERT for classification tasks. They identified class keywords as words with high attention weights during fine-tuning of a BERT classifier on class samples, thereby creating class semantic representations (CSRs). These CSRs are integrated with sentences and fed into the encoder. A matching classifier is added on top of the BERT encoder alongside a conventional K-way classifier that compares sentences with CSRs. Both classifiers are jointly trained, and CSRs are progressively improved using the updated language model. This method achieved state-of-the-art performance on text datasets, particularly in scenarios with limited labeled data.

Yang et al. [180] introduced prototype-guided pseudo-labeling (PGPL) for semi-supervised text classification. For each class, they selected the k nearest samples to the corresponding class prototype for the subsequent training iteration to ensure a balanced training process. Additionally, they trained the model with prototype-anchored contrasting, pushing samples toward their
respective class prototypes and away from others. This approach effectively alleviates underfitting near class decision boundaries and enhances text classification performance.

3.2 News Clustering

The dynamic nature of news presents a challenge for creating classifiers capable of effectively capturing and categorizing new articles. In scenarios where predefined classes are lacking and the objective is to explore the content of news articles, clustering emerges as a suitable tool. Clustering provides a robust means to organize vast collections of data without the need for predefined categories. In Paper III, we explore innovative ways to uncover patterns and structures within the ever-evolving landscape of news articles, utilizing the power of language models and clustering techniques.

3.2.1 Clustering

Clustering is a technique used in unsupervised machine learning to group similar data points together based on certain characteristics. The goal of clustering is to partition a dataset into distinct groups, or clusters, where data points within the same cluster are more similar to each other than to those in other clusters [12]. Clustering is commonly used in data analysis, pattern recognition, image segmentation, and recommendation systems, among other applications.

3.2.2 Traditional Clustering Algorithms

Many traditional strategies for clustering arbitrary sets of data points in an n-dimensional space have been proposed. These algorithms can generally be categorized into partitional, density-based, hierarchical, grid-based, and model-based categories based on their underlying principles and methodologies [173]. Each category of clustering algorithms has its own advantages and limitations, and the choice of algorithm depends on factors such as the nature of the data, the desired cluster structure, and computational considerations. In this thesis, partitional, and density-based clustering algorithms have been used.

Partitional clustering algorithms divide the dataset into a set of disjoint clusters, where each data point belongs to exactly one cluster. These algorithms typically require specifying the number of clusters in advance. Popular examples include K-Means [93] and K-Medoids [127] algorithms.

Density-based Clustering algorithms identify clusters based on the density of data points in the feature space. Clusters are formed around regions of high density, separated by regions of low density. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [32] is a popular density-based clustering algorithm.
3.2.3 Dimension Reduction

As the number of features or dimensions in a dataset increases, several challenges arise, such as sparsity of data, increased computational complexity, overfitting, and difficulty in visualization. These difficulties are collectively referred to as the curse of dimensionality [10]. Dimensionality reduction algorithms are methods utilized to decrease the number of features dimensions within a dataset while retaining its vital information. Widely employed in machine learning and data analysis, they serve to combat the curse of dimensionality [10].

Numerous algorithms are available for dimensionality reduction, falling into two main classes: matrix factorization-based methods and manifold learning-based methods. Matrix factorization methods drawn from the field of linear algebra, seek to derive a lower-dimensional representation of the data by decomposing the original high-dimensional matrix into lower-dimensional components. Popular methods in this category include Principal Components Analysis (PCA) [63], Singular Value Decomposition (SVD) [29], and Non-Negative Matrix Factorization (NMF) [77]. PCA [63], for example, is a commonly used linear dimensionality reduction technique that identifies principal components, which are directions along which data variation is most pronounced. Ultimately, PCA projects the original data onto selected principal components, effectively reducing dimensionality while preserving maximum variance.

Manifold learning methods utilize the geometric structure of the data, often represented as a neighborhood graph, to find a lower-dimensional embedding that preserves the local relationships between data points. Some of the more popular methods in this category include Spectral Embedding [9], t-distributed Stochastic Neighbor Embedding (t-SNE) [158], and Uniform Manifold Approximation and Projection (UMAP) [97]. UMAP, in particular, stands out as a state-of-the-art nonlinear dimensionality reduction approach widely adopted for visualizing and analyzing high-dimensional data. UMAP performs dimension reduction by constructing a low-dimensional embedding that captures both local and global relationships between data points, making it particularly adept at capturing complex patterns and relationships.

Furthermore, Autoencoders [48], as neural network architectures, offer a robust technique for dimensionality reduction by learning compact representations from high-dimensional data in an unsupervised manner. At their core, Autoencoders consist of two main components: an encoder and a decoder. The encoder compresses the input data into a lower-dimensional representation, while the decoder attempts to reconstruct the original input from this compressed representation. The lower-dimensional representation, also known as the latent space or encoding, captures the essential information present in the data while discarding noise and redundant features. It can then be leveraged for various downstream tasks such as data visualization, clustering, or classification.

In Paper III, PCA and UMAP have been used for dimension reduction.
3.2.4 Deep Clustering

Traditional clustering algorithms which typically assume that data is represented as feature vectors, exhibit poor performance when faced with large and high-dimensional datasets. This is primarily due to the curse of dimensionality problem [10] and the associated high computational complexity. With the remarkable success of deep learning, particularly deep unsupervised learning, various representation learning techniques have emerged in the past decade. These techniques convert unstructured data, such as text and images, into a latent space that is typically lower-dimensional and contains richer information compared to conventional feature vectors.

To cope with the challenges posed by clustering high-dimensional data, researchers have explored the use of deep representations in clustering instead of traditional feature vectors. For instance, Guan et al. [42] employed pre-trained LSTM-based text encoders, and Subakti, Murfi, and Hariadi [150] utilized BERT for text encoding. Afterwards, they normalized these representations and applied traditional clustering algorithms to them. In a related research direction, researchers have recently begun directly clustering dimension-reduced embeddings created with pre-trained language models for topic modeling [40, 189, 31] instead of relying on complex statistical models such as LDA [13]. BERTopic [40] is an example of this approach, generating document embeddings with pre-trained Transformer-based language models, clustering these embeddings, and ultimately producing topic representations using the class-based TF-IDF procedure. However, utilizing deep representations for clustering in this manner is not always optimal. These deep representations have typically been trained on general domain data for generic tasks, so they are not inherently optimized for clustering tasks and often require adaptation. In fact, deep representation learning methods struggle to integrate potential clustering information to improve the quality of learned representations, primarily due to a lack of mutual enhancement between clustering and representation learning.

To overcome these challenges, the concept of deep clustering has arisen, with the goal of optimizing representation learning and clustering simultaneously. The deep clustering model consists of a representation learning module that takes in raw data and generates a low-dimensional representation, commonly referred to as an embedding. Furthermore, it includes a clustering module that takes these low-dimensional representations as input and produces either cluster labels for hard clustering or probabilities for cluster assignment in soft clustering. The parameters of both modules are trained simultaneously using certain objective functions including clustering loss.

Initially, numerous deep clustering algorithms were introduced in the computer vision domain for clustering image datasets [177, 20, 36, 54, 55]. However, their applicability to other data types, such as text data, was constrained by the image-specific techniques employed, such as CNN architectures [76] and data augmentations. The Autoencoder [48] is a general structure that can be
customized for different data types. Hence, in the early general-purpose deep clustering approaches, Autoencoders were used as the representation learning component for different data types such as text and image data.

Xie, Girshick, and Farhadi [169] introduced the Deep Embedded Clustering (DEC) method which simultaneously updates the data points’ representations, initialized from a pre-trained Autoencoder, and cluster centers initially computed through K-Means clustering. They introduced the self-training technique to the deep clustering task, initiating an active branch of methods referred to as self-training deep clustering. More specifically, the model’s parameters are optimized by minimizing the KL-divergence [70] between the soft cluster assignments and an auxiliary distribution derived from these assignments. The assignment distribution $Q$ for each data point $x_i$ is determined by calculating the Student’s t-distribution [149] similarity between the data point representation $h_i$ and the cluster centroids:

$$q_{ik} = \frac{(1 + \|h_i - \mu_k\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_j^K (1 + \|h_i - \mu_j\|^2/\alpha)^{-\frac{\alpha+1}{2}}}$$

where $q_{ik}$ represents the probability of instance $x_i$ belonging to cluster $k$, $K$ is the total number of clusters, $\mu_k$ denotes the representation of cluster $k$, and $\alpha$ signifies the degree of freedom of the Student’s t-distribution. The auxiliary distribution $P$ is a modified version of the assignment distribution $Q$ computed as follows:

$$p_{ik} = \frac{q_{ik}^2/f_k}{\sum_j^K q_{ij}^2/f_j}$$

where $f_k = \sum_i^N q_{ik}$ are soft cluster frequencies. Raising $q_i$ to the second power helps the model prioritize learning from instances with higher confidence, effectively reducing the influence of low-confidence instances during training. Moreover, normalizing by frequency per cluster regulates the contribution of clusters with varying sizes in the loss function, thereby mitigating the risk of degenerate solutions where all instances are assigned to a single cluster.

Accordingly, the objective function utilized for training the deep clustering model is computed as follows:

$$L = KL(P \parallel Q) = \sum_i \sum_k p_{ik} \log \frac{p_{ik}}{q_{ik}}$$

It is worth mentioning that DEC utilized TF-IDF vectors as text features for the input of the Autoencoder. DEC is highly significant in deep clustering and has been employed as a baseline in numerous studies including our research presented in Paper III.

Some studies have explored integrating various clustering losses or additional loss functions into the optimization process. IDEC was subsequently introduced by Guo et al. [43] as an enhancement to DEC, incorporating the Autoencoder’s reconstruction error into the objective function. Deep K-Means
[104] employs a general form of the K-Means objective function along with the Autoencoder reconstruction loss, as an alternative to the DEC loss function.

Moreover, some studies tried to customize DEC for text data. Instead of TF-IDF vectors used in DEC, Hadifar et al. [44] utilized Smooth Inverse Frequency (SIF) embeddings [5] which were considered more suitable representations for short text than TF-IDF. Motivated by the great success of PLMs in NLP, Huang et al. replaced the Autoencoder component of the DEC architecture with BERT [28], fine-tuning it simultaneously with masked language modeling loss and DEC clustering loss.

It is important to note that while Autoencoders succeed at dimensionality reduction, their primary focus may limit their ability to comprehensively capture the underlying data distribution within the latent space. Specifically, these representations are learned in an instance-wise manner, overlooking the interrelations among different instances. Consequently, the resulting embeddings may fail to effectively discriminate between instances in the embedding space, thereby leading to suboptimal clustering performance.

Inspired by the success of contrastive representation learning, contrastive learning has also been introduced into deep clustering. Contrastive learning has emerged as a highly popular unsupervised representation learning technique in recent years. Its fundamental principle involves bringing positive pairs of instances closer together while pushing negative pairs further apart, a concept often referred to as instance discrimination. At the core of contrastive learning lies the Normalized Cross-Entropy with Information Maximization (InfoNCE) loss [108] which for a set of $N$ random samples is formulated as:

$$ L_{InfoNCE} = -\log \sum_{i=1}^{N} \frac{\exp(f(h_i, h_i^\tau)/\tau)}{\sum_{j=1}^{N} \exp(f(h_i, h_j^\tau)/\tau)} $$

where $h_i$ is the representation of anchor sample, $h_i^\tau$ and $h_j^\tau$ are the representations of the positive and negative samples respectively, $f$ is a similarity function such as cosine similarity, and $\tau$ is a temperature parameter that controls the smoothness of the probability distribution. Positive samples are typically generated through data augmentation, which may vary depending on the data type and the specific task at hand. The vision-language model CLIP [123] is one of the very successful applications of contrastive learning, where it is used to learn a joint representation of images and text.

Similar to contrastive representation learning, the objective of contrastive deep clustering is to pull positive pairs closer while pushing the negative pairs away. However, the distinction lies in how positive and negative pairs are defined. Similar to other deep clustering methods, contrastive deep clustering has its origins in the field of computer vision. SCAN [159], GCC [191], SwAV [19], MiCE [157], and LNSCC [92] have recently demonstrated state-of-the-art clustering performance on image datasets through contrastive learning techniques. SCAN utilizes the nearest neighbors in the k-Nearest Neighbors (kNN) graph, suggesting that a sample and its nearest neighbors should be grouped into the
same cluster. In contrast, GCC assumes that the transformation of an image and its neighbors’ transformation should exhibit similarity, thereby enhancing clustering performance on image data. Based on the insight that neighboring samples in a kNN graph might not consistently share the same category and that distinguishing between positive and negative pairs can be challenging in a naïve kNN setup, LNSCC proposes a soft contrastive clustering approach. This method assigns positivity and negativity scores to each pair of samples to capture their similarity and dissimilarity, thereby addressing ambiguity at cluster boundaries and yielding clearer distinctions between clusters.

Researchers in NLP have also explored the integration of contrastive learning into text deep clustering. SCCL [185] was among the first to utilize instance-level contrastive learning in text deep clustering. It employs Sentence-BERT [129] as the text encoder and combines the DEC [169] clustering objective with the contrastive InfoNCE loss [108] for optimizing the model parameters. However, while instance-level contrastive learning is successful at learning general feature representation, it overlooks semantic-level correlations within the same cluster, leading to suboptimal clustering outcomes and sparse cluster spaces. DACL [81] tackles this issue by smoothly shifting the loss weight of the model from contrastive learning to clustering throughout training and filtering negative samples in contrastive learning using pseudo-labels generated by clustering.

Pseudo-labeling, already discussed in Section 2.3.3 in the context of teacher-student models, has recently found its way into the domain of deep clustering. The proposed methods typically involve an iterative process where a clustering module and a classification module mutually enhance each other, resulting in notable performance gains. For instance, DeepCluster [18] employs an iterative approach wherein image features extracted by a convolutional neural network are clustered using a standard algorithm like K-Means. The resulting assignments, serving as hard pseudo-labels, are used for updating network’s weights. Pseudo labeling has extended the capabilities of semi-supervised learning to unsupervised clustering tasks. However, its effectiveness heavily depends on the quality of the pseudo-labels used for training the classifier, which are influenced by model capacity and hyperparameter tuning. While existing methods [107, 159] have addressed this challenge by incorporating pre-training as an initial step before pseudo-labeling, further attention is needed in this area.

The exploration of pseudo-labeling in deep clustering for text data has been limited. Rakib et al. [126] proposed an iterative method where a Multinomial Logistic Regression classifier is trained using cluster labels from non-outlier samples. This classifier is then employed to correct the clustering outcome by reclassifying outliers, with the resulting set of clusters serving as input for the next iteration. However, this approach relies on fixed TF-IDF representations for clustering, potentially limiting its generalizability.
Chapter 4

Event-based Topic Discovery in News Streams

This chapter provides the groundwork for Papers IV and V, which tackle research questions RQ3 and RQ4, respectively. Paper IV introduces a novel model for story discovery in news streams. Paper V explores the relationship between news text and images and introduces a multimodal dataset for event-based topic discovery in multimodal news streams, along with a baseline model tailored for this task.

4.1 Topic Detection and Tracking

Topic Detection and Tracking (TDT) is a prevalent technique in the field of information retrieval (IR) and is pivotal for exploring, mining, and organizing news stories across various media sources. Introduced by Allan [1], TDT aims to identify and monitor real-world events within a multi-source news stream. In the context of TDT, a news story is a report on a particular event, and a topic is characterized by a collection of news stories discussing different aspects of the same event. When a plane crashes in Malaysia, it serves as the seminal event that initiates the topic. Any stories detailing the crash cause, death toll, rescue efforts, survivors, and so on are all considered part of the topic. Stories covering a separate plane crash in a different country on the same day or an earthquake in Japan would not typically fall under the same topic. However, in some instances in the literature, the terms news story and topic have been used interchangeably.

It is essential to distinguish an event topic from the conventional notion of topic found in information organization research. While the latter typically embodies the theme or subject of a text, such as “sports” or “politics” in news classification, event topics focus specifically on the triggering event of a story. Furthermore, topics in TDT evolve over time and may encompass stories that
are not necessarily related in subject matter.

TDT comprises five core tasks. *Story Segmentation* involves breaking down continuous news texts, such as transcriptions of news shows, into individual stories. However, this task becomes meaningless when processing streamed news from websites where stories are already separated. *First Story Detection* focuses on identifying stories that are not associated with previously recognized events, potentially signaling the beginning of a new event topic. *Cluster Detection* aims to allocate new stories within a streaming dataset to relevant topics in real-time, either by linking them to existing topic clusters or by creating new ones as needed. *Tracking* is closely linked to cluster detection and involves monitoring existing topics while continually seeking out additional stories to enrich them over time. *Story Link Detection* involves determining whether two given stories are related and belong to the same topic or not. Some of these tasks are closely interconnected, and their collective contributions enable the functionality of TDT. However, numerous studies in the literature often tackle these tasks within their proposed methods without explicitly specifying individual components for each task.

In the literature, TDT has been framed as a non-parametric topic modeling problem [192] which falls outside the scope of this thesis. Alternatively, TDT has been approached as a stream clustering problem. These works refer to this task as *online story discovery* or *news stream clustering*. It is noteworthy that in these studies, the terms *event topic* and *news story* are used interchangeably, deviating slightly from their definitions in the TDT task outlined by Allan [1].

Early attempts at news story discovery relied on sparse document representations such as keywords and TF-IDF vectors. Laban and Hearst [72] extracted article keywords and constructed a graph of articles spanning a window of $N$ days such that articles sharing more keywords than a specified threshold were connected. Local topic clusters were then identified using the Louvain community detection algorithm [14]. The window was moved along the news stream, and if a topic continued to receive new articles across overlapping graphs, all of these articles were linked to the same topic. This process enabled the story to develop and grow over time. For longer-term stories, topics from non-overlapping windows were combined if their similarity exceeded a certain threshold. Staykovski et al. further improved this approach by utilizing TF-IDF vectors instead of keywords.

Miranda et al. [102] investigated a multilingual news stream. Their monolingual article representation comprised TF-IDF subvectors for words, word lemmas, and named entities extracted from different document sections: the title, the body, and a combination of both, totaling nine subvectors. Additionally, they developed a cross-lingual version of these vectors. Their methodology involves computing similarities between the monolingual TF-IDF subvectors of an article and those of monolingual clusters, which are the aggregated subvectors of their members. These similarities are then aggregated using a Rank-SVM model. The decision to merge the document with an existing cluster or create a new cluster is determined by another SVM classifier. Both SVM
models undergo training using a supervised training set. Furthermore, they incorporated article timestamps to prevent recent documents from merging with older clusters. In this setup, a crosslingual cluster comprises several monolingual clusters in different languages. During stream processing, after updating monolingual clusters, adjustments are made to crosslingual clusters accordingly.

With the emergence of dense document representations containing richer semantic information, researchers have begun exploring their potential in news story discovery. Staykovski et al. [147] conducted a comparison between TF-IDF and doc2vec representations for this purpose and concluded that sparse representations are more effective. In a more recent study, Saravanakumar et al. [137] adopted a methodology similar to that of Miranda et al. [102] for news story discovery but they used a combination of sparse and dense representations for articles. They demonstrated that incorporating contextual BERT representations alongside TF-IDF representations could enhance performance in this task. This enhancement was achieved through fine-tuning BERT on event similarity using a triplet network architecture [53] and incorporating external entity knowledge.

The weaker performance of dense representations like BERT (without being fine-tuned) in comparison to sparse representations, in news story discovery may be attributed to the low uniformity of their embedding space. Alignment and uniformity, as discussed in [160], are fundamental attributes of any embedding space. For the task of news story discovery, alignment refers to how closely articles related to the same story are positioned within the embedding space, while uniformity is a measure of the uniform distribution of random articles throughout that space. Lack of uniformity poses a challenge in distinguishing between two articles that share a common theme but pertain to different events.

In recent years, contrastive learning has proven highly effective across various language processing and computer vision tasks. This effectiveness primarily arises from its capacity to improve the alignment and uniformity of embedding spaces, as demonstrated by Wang and Isola [160]. A notable example of this success in news story discovery is shown in the study by Yoon et al. In this work, with the idea that not all sentences in the article have the same significance for its story, a story-indicative article representation is made by aggregating the sentence representations, derived from a pre-trained Sentence-Transformer, via a single transformer layer. Subsequently, these representations are compared with existing cluster representations within the current window to either identify the best match or create a new cluster. Once clusters are defined, the representations are further refined to adapt to the recent context through cluster-level contrastive learning. This research demonstrated that these dense representations outperform sparse alternatives.

Despite numerous efforts to discover effective article representations that facilitate the seamless identification of news stories and differentiate between various stories, this remains an active research domain.
4.2 Online Clustering

Another aspect worth exploring in previous studies on news story discovery is the clustering methodology employed. Given that TDT is an online task and articles in the news stream require real-time clustering, opting for an online clustering algorithm is a natural choice. However, there is variation among previous works regarding their approach to online data processing. Many of them favor a non-parametric version of online K-Means clustering.

The online K-Means algorithm is a modification of the traditional K-Means algorithm, allowing for continuous learning and cluster updating as new data points emerge over time. In the non-parametric version, the number of clusters is not fixed and can expand indefinitely. This characteristic aligns well with the demands of story discovery, as the clustering problem is inherently non-parametric, and each document in the stream could potentially initiate a new event cluster.

Here is a breakdown of how the non-parametric online K-Means operates:

- **Data Streaming**: instead of processing the entire dataset simultaneously, the online K-Means algorithm handles data points one at a time as they are received.

- **Assignment**: upon receiving a data point, the algorithm assigns it to the nearest centroid based on a chosen distance or similarity metric, such as Euclidean distance or cosine similarity. Alternatively, if the data point is not sufficiently close to any existing cluster, the algorithm creates a new cluster for it. In the literature, this decision has been made through both supervised and unsupervised approaches. For instance, Miranda et al. [102] and Saravanakumar et al. [137] utilized trained classifiers to determine when a new cluster should be created, using labeled training datasets. Alternatively, a similarity threshold can be employed, with optimal values determined via grid search if supervised data is available. However, obtaining supervised data is not always feasible due to the high cost associated with acquiring human annotations and the challenge of keeping it up-to-date. Therefore, an unsupervised approach is often more practical and suitable for evolving news article streams.

- **Centroid Update**: following the assignment of a data point to a cluster, the algorithm adjusts the centroid of that cluster to incorporate the new data point and adapt to the evolving data distribution.

Algorithm 1 shows the pseudocode of this clustering algorithm. As previously emphasized, time plays a crucial role in news stories. Some studies employing this online clustering algorithm [102, 137], incorporated timestamp features for each cluster. During document-cluster comparisons, such methods assess not only textual representations but also timestamp features, making decisions based on a combination of these comparisons. A significant disparity between the publishing time of an article and the timestamp features of a
Algorithm 1: The non-parametric online K-Menas clustering algorithm for data stream clustering

Data: \( \mathbb{D} \): a news article stream
\( f \): document representation generation function
\( \theta \): article-story similarity threshold

Result: A set \( \mathbb{S} \) of stories in stream \( \mathbb{D} \)

1. \( \mathbb{S} \leftarrow \emptyset \)
2. for every new article \( d \in \mathbb{D} \) do
   3. \( R_d \leftarrow f(d) \)
   4. if \( \max(\{\text{sim}_{d,s_j} | s_j \in \mathbb{S}\}) > \theta \) then
      5. Assign article \( d \) to corresponding \( s_j \)
      6. Update \( R_{s_j} \) with \( R_d \)
   7. else
      8. \( s_{|\mathbb{S}|+1} \leftarrow \{d\} \)
      9. \( R_{s_{|\mathbb{S}|+1}} \leftarrow R_d \)
      10. \( \mathbb{S} \leftarrow \mathbb{S} \cup \{s_{|\mathbb{S}|+1}\} \)
   11. end
3. end
4. return \( \mathbb{S} \)

cluster serves as a valuable indicator that the article may not belong to that cluster, even if the textual features exhibit a reasonable match between the incoming article and the cluster representation. Yoon et al. [182] adopted a different approach, employing a sliding window mechanism along the stream. In this approach, documents are compared only with the active clusters within the time frame of the sliding window, eliminating the need to explicitly consider temporal features for the clusters. Moreover, incorporating the window to the online algorithm enhances the algorithm’s efficiency and speed for large-scale datasets with numerous topics. In Paper IV and Paper V, we also utilize this approach for online clustering of news articles within the news stream.

Additionally, some studies in the literature employed a two-step clustering approach. They analyze a collection of articles gathered over a specific time frame, such as \( N \) consecutive days, to form local clusters. Subsequently, these local clusters are linked over time to track the progression of stories. For instance, Laban and Hearst [72] and Staykovski et al. [147] construct a graph of articles spanning a window of \( N \) days based on the similarity of article representations. They then apply a Louvain community detection algorithm [14] to identify local topics within the current window. As the window moves along the news stream, if a topic consistently receives new articles across overlapping windows, all these articles are associated with the same topic. This mechanism facilitates the gradual development and expansion of the story over time. For longer-term stories, topics from non-overlapping windows are compared based on their keyword distributions, and if their similarity surpasses a certain threshold, they are merged. Linger and Hajaiej [84] introduced a similarity-
based *replaying* strategy to connect local topics into cohesive stories. For a new batch of articles at time $t$, they calculate similarities between all new articles and all topics from the previous time $t-1$. If a topic from $t-1$ exhibits a similarity with a new article at time $t$ surpassing a predefined threshold, all articles associated with that topic are included in the current batch. This allows them to be considered during the subsequent round of topic detection at time $t$. These two-step algorithms do not utilize explicit time features; rather, time is implicitly incorporated through the batch/window procedure.

Unlike algorithms that process articles one by one, batch processing algorithms are better suited for handling large-scale streams where scalability is a key concern. Additionally, they facilitate the emergence of various topic behaviors such as splitting and merging over time. However, detecting such topic behaviors across batches/windows and tracking stories may pose challenges and add complexity to the stream clustering model.

### 4.3 Multimodal News Streams

News websites have evolved to incorporate a diverse array of presentation modalities, such as text, images, diagrams, and videos, strategically designed to engage readers and convey messages effectively. Each modality offers unique advantages and constraints, and their integration can enrich the user experience, making it more immersive and engaging. In this section of the thesis, *multimodal* specifically refers to the combination of images and text, while other modalities are not considered within the scope of this study.

Analyzing multimodal news poses significant challenges due to the varied interrelationships between information from different modalities. News article texts typically contain an abundance of details ranging from timing and content to location and individuals involved in reported events. In contrast, the role of accompanying images in news articles is diverse. Images may serve as decorative elements, provide supplementary information, or, at times, present potential sources of misinformation. For instance, imagine a news article highlighting a specific action by Trump, accompanied by an image solely featuring Trump himself.

Studying the relationship between text and images, especially in the context of news analysis, is an interdisciplinary research question that has garnered significant attention from various fields, including communication science, media studies, journalism, machine learning, and multimodal analysis. Several taxonomies of image-text relations, sometimes specifically for analyzing news articles, have been proposed in media studies [8, 17] and semiotics [7, 94, 95]. Among these, Barthes’ work [7] stands out as pioneering. He categorizes text-image relations into three main types: (1) Anchorage, where text describes the image; (2) Illustration, where the image visually represents information from the text; and (3) Relay, where text and image share an equal relationship, such as complementarity or interdependence.
However, there is limited work in computational approaches for multimodal news analysis that attempts to model and utilize the relationship between images and text. Müller-Budack et al. [105] introduced an unsupervised approach that measures the cross-modal consistency of entity relations between image and text modalities in news articles. Oostdijk et al. [109] investigated the relationship between text and images in news articles for flooding-event detection. They identified four cross-modal relations: images visualizing what the text describes, images visualizing people referred to in the text, images visualizing a situation as it existed before while the text describes or suggests how a similar situation might arise (flood threat), and images visualizing a situation as it exists now but which will be affected by developments described in the text (e.g., an image of an elephant in an area that will be flooded once a dam is ready). Nonetheless, the scope of their investigation is restricted, and its generalizability might be limited. In a recent study, Cheema et al. [22] introduced a framework for the computational analysis of multimodal news. Drawing from real examples of news reports, they outlined a set of image-text relationships and multimodal news values, exploring their implementation through computational methods. Yet, there has been no research exploring the relationships between images and text for story discovery in news streams, so the extent to which multimodal information aids in the online story discovery task remains an open question.

Prior research in multimodal news analysis has primarily concentrated on two main areas: thematic classification of news [161, 65, 118] and fake news detection [168, 195, 194, 179]. The only instance of multimodal work in topic detection and tracking, to our knowledge, is by Li et al. [82], who specifically explored topic detection and tracking within video news.

4.3.1 Deep Learning for Multimodal Data

Advancements in multimodal deep learning have empowered vision-language models such as CLIP [123], BLIP2 [80], and LLaVA [87] to comprehend fundamental relationships between modalities, such as correlations between words and phrases and their visual representations. While these developments have fueled significant progress in tasks like image captioning, text-to-image generation, and visual question answering, they are inadequate for generating multimodal representations for complex objects like multimodal news articles, thereby restricting their capacity to interpret the overall multimodal message.

Many existing studies in multimodal news analysis utilize diverse fusion models to integrate image and text, creating a multimodal representation [161, 65, 118, 168, 194, 100]. Typically, these studies employ modality-specific encoders to generate embeddings for each modality. These embeddings are then projected into a shared space to enable comparison between different modalities before being fused for the downstream task.

Various fusion approaches, including early fusion and late fusion, have been proposed to leverage heterogeneous data and modalities. Early fusion, also
known as feature-level fusion, aggregates all features, including textual and visual features, into a single feature vector which serves as the multimodal representation. This can be achieved through concatenation [161, 65], or employing attention mechanisms [118, 168, 194, 100]. The resulting representation is then used for downstream tasks. In late fusion, modalities are merged at the decision level. In classification tasks, this usually entails combining the posterior probabilities derived from classifiers for each class. Rather than directly predicting labels, these classifiers produce probabilities for various classes [83].

It is worth mentioning that CLIP [123] has been utilized in literature to measure cross-modal similarity [195, 194]. CLIP, a multimodal model trained on diverse image-text pairs, is capable of predicting relevant text snippets for given images and vice versa. This integration enables CLIP to embed texts and images into a unified latent space, facilitating the calculation of cross-modal correlations. Consequently, the cosine similarity between CLIP representations of text and image modalities indicates the extent to which the article text and image are aligned, serving as a criterion to adjust the contribution of the image modality in the overall multimodal representation of the article. This technique has been used in Paper V.
Chapter 5

Summary of Contributions

This thesis aims to answer five pivotal research questions:

**RQ1** How can news topics be automatically identified across various granularity levels?

**RQ2** What effective methodologies can be employed to integrate deep learning into the investigation of news topics when labeled data is scarce or unavailable?

**RQ3** How can deep learning techniques be utilized for topic identification in news streams while effectively addressing challenges associated with changes in topic focus and evolution over time?

**RQ4** What is the interrelation between different modalities within multimodal news, and how can these modalities be harnessed for the purpose of topic identification?

In response to **RQ1**, the thesis proposes classification and clustering for coarse-grained topic identification and event-topic discovery in news streams for fine-grained topic identification. Additionally, it addresses **RQ2** by proposing semi-supervised deep classification and deep clustering approaches for topic identification in cases where supervision is limited or absent, respectively. **Paper I**, **Paper II**, and **Paper III** focus on addressing research questions **RQ1** and **RQ2** for coarse-grained topics. **Paper I** and **Paper II** propose the development of semi-supervised classification models using deep learning for news topic identification in cases where pre-defined coarse-grained topics are of interest and there is insufficient labeled data available to effectively train a classifier in a fully supervised manner. Additionally, **Paper III** proposes deep clustering in scenarios where a set of predefined classes is absent, yet there is a desire to explore coarse-grained topics within the news dataset.

**Paper IV** and **Paper V** address research questions **RQ3** and **RQ4**, respectively. **Paper IV** introduces a novel model for story discovery in news
streams, while Paper V initiates the study of using both news text and images. In the discovery of event-based topics, in particular, it introduces a multimodal dataset for event-based topic discovery in multimodal news streams, along with a baseline model tailored for this task.

5.1 Paper I


Paper Contributions

This paper is motivated by a scenario in contextual advertising where the number of classes is known, but there are only a few labeled examples available for each class, while the majority of the dataset remains unlabeled. To tackle this issue, the paper introduces Cformer, a semi-supervised approach that utilizes the teacher-student architecture employed in pseudo-labeling.

Cformer adapts the MPL method proposed by Pham et al. [115] from the computer vision field for semi-supervised text classification, incorporating necessary modifications to suit text data. This architecture aims to mitigate the confirmation bias inherent in pseudo-labeling methods by iteratively training the teacher and student models. Feedback from the student to the teacher in each iteration informs the teacher about the quality of the generated pseudo-labels, facilitating self-improvement. Consequently, the teacher is trained using a supervised loss computed for the labeled dataset, a consistency loss calculated based on the unlabeled dataset and an augmented version of it, and feedback from the student, represented by the loss value of the student for the labeled dataset. Additionally, the student undergoes supervised training using the pseudo-labels generated by the teacher for the unlabeled data.

In Cformer, the teacher and student share the same architecture, which consists of a BERT encoder followed by an MLP for performing the classification task. Furthermore, the paper proposes a version of Cformer called Distill-Cformer, in which a DistilBERT model is used as the text encoder in the student. After being trained, this student is better suited for resource-limited environments.

The experiments demonstrated that Cformer could surpass state-of-the-art semi-supervised text classification methods when a reasonable amount of labeled data for each class is available. Additionally, despite its smaller size, Distill-Cformer exhibited performance on par with Cformer.
Author Contributions

As the main author, I contributed to formulating the problem, implementing the code and experiments, analyzing the results, and leading the writing of the first draft. Xuan-Son Vu offered valuable guidance and support throughout the process, especially in formulating the problem, analyzing the results, and incorporating them into the first draft. Frank Drewes fulfilled advisory roles, engaging in discussions concerning problem formulation, experiments, and result presentations. Additionally, he made significant contributions to the writing of the first draft by writing the introduction section and reviewing and providing feedback on other sections. Monowar Bhuyan engaged in discussions and provided feedback on the draft.

5.2 Paper II

Arezoo Hatefi, Xuan-Son Vu, Monowar Bhuyan, and Frank Drewes. The Efficiency of Pre-training with Objective Masking in Pseudo Labeling for Semi-Supervised Text Classification. Submitted to the Northern European Journal of Language Technology (NEJLT), 2023.

Paper Contributions

This paper proposes CformerM, an extension of the Cformer introduced in Paper I. CformerM incorporates an unsupervised pre-training phase, further training the text encoders of the teacher and student models on the unlabeled data using objective masking. Objective masking prioritizes masking topic words from a topic word list, supplemented by random word masking if necessary, to mask a total of 15% of the words of the text. This masking objective aims to enhance the text encoder ability to grasp the underlying topics in the dataset and recognize its topical information.

To create the topic word list, the dataset undergoes LDA \cite{13} topic modeling with an appropriate number of topics. The number of topics is selected based on the coherence scores of various topic models with differing numbers of topics. Then, the $N$ most relevant words for each topic are extracted using the relevance measure introduced by Sievert and Shirley \cite{142} and compiled into a list. This measure includes a parameter $\lambda$ that allows for the selection of the specificity of the topic words. When $\lambda$ is small, the method prioritizes words strongly associated with the topic but less common in other topics, resulting in distinct topics but potentially neglecting relevant words shared across topics. Conversely, with a higher $\lambda$, the approach concentrates on words prevalent within the topic and also across other topics, capturing more general aspects of the topics. Additionally, the optimal value for $N$ is determined through assessing the coherence of different lists with varying values for $N$.

In extensive experiments conducted on datasets in English and Swedish, CformerM was compared with numerous baselines, including Cformer, various
state-of-the-art semi-supervised classifiers, and a variant of CformerM achieved by employing random masking instead of objective masking. The experimental results indicated that CformerM outperforms Cformer and other baselines in most cases across all datasets. However, the influence of objective masking on classification accuracy is more notable when the amount of supervised data for classification is limited.

In comparing CformerM and its variant created with random masking, it was demonstrated that when the dataset significantly deviates from the BERT training data and includes domain-specific information, such as medical documents, the difference in the impact of objective masking and random masking on the classification performance becomes more noticeable.

Moreover, a comparison was made between the proposed LDA-based method for generating topic word lists and a simpler technique that uses TF-IDF to identify topic words within the corpus. Specifically, words are sorted based on their average TF-IDF scores across all documents, and the top words are selected. It was found that creating the topic word list based on TF-IDF instead of LDA is less effective, particularly when the labeled data is severely limited. It was hypothesized that this superiority of CfromerM could be attributed to the fact that the topic model considers the underlying structure of the dataset, whereas TF-IDF relies on individual documents. Additionally, the LDA-based method offers flexibility in choosing between highly topic-specific words and more general ones, addressing the specific needs of the analysis, while the TF-IDF method offers less control over the generated lists.

Last but not least, a qualitative analysis conducted indicated that pre-training with objective masking enhances the reliability and interpretability of the model, resulting in more accurate classification results. Additionally, experiments conducted in a zero-shot setting demonstrated that the proposed pre-training of the language model with objective masking could enhance the language model’s ability to recognize examples of classes that had not been seen before.

Author Contributions

As the main author, I contributed significantly to formulating the problem, implementing the code and experiments, analyzing the results, and leading the writing of the first draft. Xuan-Son Vu provided valuable guidance and support throughout the process, particularly in formulating the problem, analyzing the results and incorporating them into the first draft, and enhancing the illustrations. Frank Drewes fulfilled advisory roles, engaging in discussions concerning problem formulation, experiments, and result presentations. Additionally, he made significant contributions to the writing of the first draft by writing the introduction section and reviewing and providing feedback on other sections. Monowar Bhuyan engaged in discussions and provided feedback on the draft.
5.3 Paper III


Paper Contributions

Paper III introduces ADCluster, a deep clustering approach based on pseudo-labeling, for document clustering. ADCluster comprises a clustering and a classification component that iteratively promoted each other, leading to significant performance improvements. During each iteration, K-Means clusters the document representations generated via the language model and predicts pseudo-labels. Subsequently, these labels are utilized to train the classifier, consisting of the language model encoder followed by an MLP for classification, in a supervised manner. This iterative adaptation, referred to as inner adaptation, allows the PLM to adjust to the clustering task and generate more clustering-friendly representations, thereby enhancing K-Means clustering in subsequent epochs.

The paper also explores the adaptation power of ADCluster over time to growing sets of documents, a process referred to as outer adaptation. Outer adaptation resumes the inner adaptation when a significant amount of new data becomes available, either by considering the entire dataset (accumulative outer adaptation) or using only the new data (non-accumulative outer adaptation). In this dynamic setting, the assumption is made that the number of clusters over time remains constant, with new samples being received. In this scenario, distribution shift only occurs within clusters. This setup is motivated by a scenario in which there is a steady stream of content, such as news articles, centering around a fixed set of topics, albeit with changing focus over time. For instance, during major sports events like the FIFA World Cup, sports news primarily revolves around this event, even though this may not have been the case previously.

Extensive experiments conducted on various short and long text datasets demonstrated ADCluster’s superiority over established document clustering techniques, particularly on medium and long-text documents, by a significant margin. Furthermore, the proposed approach surpassed well-established baseline methods in both accumulative and non-accumulative outer adaptation scenarios.

Author Contributions

As the main author, I contributed to formulating the problem, implementing the code and experiments, analyzing the results, and leading the writing of
the first draft. Xuan-Son Vu provided valuable guidance and support throughout the process, particularly in formulating the problem and improving the first draft particularly the algorithm and illustrations. Frank Drewes fulfilled advisory roles, engaging in discussions concerning problem formulation, experiments, and result presentations. Additionally, he made significant contributions to the writing parts of the first draft and reviewing and providing feedback on the other parts. Monowar Bhuyan engaged in discussions and provided feedback on the draft.

5.4 Paper IV


Paper Contributions

The paper introduces a methodology for discovering news stories within a news stream known as PromptStream. PromptStream utilizes an online clustering approach to assign articles in the stream to their relevant stories. This involves employing a sliding window that traverses the stream, serving as a representation of the time frame of interest. As the window progresses, new news articles are sequentially clustered into news stories based on their temporal order. To assign a news article to a suitable cluster, PromptStream compares it only with the existing clusters within the time frame of the sliding window. If the article’s resemblance to a story exceeds the similarity threshold, it is clustered into that cluster; otherwise, a new story is initiated with this news article.

PromptStream generates topic-aware document representations by combining a prompt-based representation with the output of a mean pooling layer applied to the last layer of the PLM. The prompt-based representation is constructed using a cloze-style template that prompts the model about the topic of the given text:

```
[ topic : <mask> ] <title> <body>
```

where <title> and <body> represents the title and body of the news article, respectively. This representation extracts topic-specific information from the text by prioritizing attention to topic-related tokens and entities. Conversely, mean pooling provides a broader representation of the entire document. By integrating these two representations, the approach effectively leverages both the detailed, contextually rich information acquired from cloze-based prompting and the global context captured through mean pooling.
Moreover, the text encoder remains consistently updated to reflect the latest context within the news stream through continual learning techniques. A memory is maintained, filled with the most confident clustering results based on the resemblance of the articles to the stories they are clustered into for a certain duration (e.g., 10 days). At the end of this period, these samples are replayed to update the encoder using cluster-level contrastive learning. This process encourages articles to move closer to the center of their respective clusters while simultaneously being pushed away from other cluster centers, resulting in enhanced uniformity and alignment of the embeddings. Given a batch \( B \) of positive article-story pairs \((d, s) \in B\) the cluster-level contrastive loss function is computed as follows:

\[
L_{cts} = - \sum_{(d, s) \in B} \log \frac{\exp \left( \frac{\cos(R_d, R_s)}{\tau} \right)}{\sum_{s' \in S_W} \exp \left( \frac{\cos(R_d, R_{s'})}{\tau} \right)}
\]

where \( \tau \) is a temperature parameter and \( S_W \) is the set of existing stories in window \( W \). Through extensive experiments, PromptStream was compared with state-of-the-art methods, demonstrating its superior performance across three news stream datasets.

**Author Contributions**

As the main author, I played a vital role in formulating the problem, implementing the code and experiments, analyzing the results, and leading the writing of the first draft. Anton Eklund engaged in discussions concerning problem formulation and experiment designs. Additionally, he conducted a qualitative analysis on the method’s results, presenting them in the “Qualitative Analysis” section of the paper. Moreover, he made significant contributions to improving the first draft and enhancing the illustrations. Mona Forsman fulfilled advisory roles, engaging in discussions concerning problem formulation, experiments, and result presentations. Additionally, she reviewed and commented on the draft.

5.5 Paper V

Arezoo Hatefi, Johanna Björklund, Xuan-Son Vu, and Frank Drewes. METOD: A Dataset and Baseline for Multimodal Discovery of Event-Based News Topics. *Submitted to the International Journal of Multimedia Information Retrieval*, 2024.

**Paper Contributions**

Given that online news reporting typically integrates various modalities such as text, images, video, audio, and other data types to convey information, this paper proposes event-based topic discovery in a stream of multimodal news
articles as a significant and challenging problem within the broader field of topic discovery. To address the lack of an appropriate dataset for this task, the authors annotated a dataset of image-text news articles from the New York Times, named METOD, enabling researchers to develop and evaluate methods for this task.

Event-based topics typically have a limited lifespan. For instance, in the case of a sudden event like an earthquake, the initial articles usually appear shortly after the event, and coverage gradually diminishes over time. Considering the temporal aspect of event-based topics, this paper defines some characteristics for such topics that could be used for evaluating the performance of event-based topic discovery algorithms. These characteristics include topic size, topic duration, article frequency, temporal irregularity, disconnection index, suddenness, specificity, and image informativeness. Except for the last characteristic, others are relevant for only-text news streams as well. Additionally, the values of these characteristics are computed for the topics in the METOD dataset to the extent possible.

Moreover, the Multimodal EventTracker, a baseline model for event-based topic discovery in multimodal news streams, is introduced and its performance on the METOD dataset is analyzed. Multimodal EventTracker bears similarity to PromptStream introduced in Paper IV from the online clustering aspect. However, its encoder differs in that it is tailored to produce a robust representation for text-image data. Additionally, the encoder remains fixed and does not undergo continual learning. To generate the representation for the image-text data, both text and image are initially encoded using text-specific and image-specific encoders. Subsequently, they are combined with the similarity of the text and image representations generated with CLIP [122] serving as the weight of the image representation.

**Author Contributions**

I, Johanna Björklund, and Frank Drewes contributed equally to the problem formulation, conceptualization, paper writing, and dataset development. In addition, I developed and implemented the baseline model, and designed and conducted the experiments. Moreover, Xuan-Son Vu engaged in the discussions regarding problem formulation, and design and implementation of the baseline model. He also made the first version of Figure 1, wrote the “Datasets in news clustering” part of the “Related Work” section, and reviewed and commented on the other parts of the manuscript.
Bibliography


