smart cities

MDPI

Article

An Unsupervised Integrated Framework for Arabic Aspect-Based Sentiment Analysis and Abstractive Text Summarization of Traffic Services Using Transformer Models

Alanoud Alotaibi [1,2,*] and Farrukh Nadeem [1]

[1] Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia; fabdullatif@kau.edu.sa
[2] Department of Information Systems, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11623, Saudi Arabia
* Correspondence: amdotibi@imamu.edu.sa

Highlights

What are the main findings?

- The integrated ABSA-ATS framework combining BERTopic and fine-tuned CAMeL-BERT models achieved 92% accuracy in aspect-sentiment classification and state-of-the-art ROUGE-L score of 0.79 for Arabic traffic services summarization, surpassing previous benchmarks by over 24 percentage points.
- Domain-specific FastText embeddings significantly outperformed general-purpose Arabic language models, identifying 16 distinct traffic service categories while demonstrating superior outlier handling and topic clustering capabilities.

What is the implication of the main finding?

- This framework enables traffic agencies to automatically analyze and summarize large volumes of Arabic social media feedback without extensive manual annotation, providing actionable insights to improve service quality and inform policy decisions.
- The unsupervised approach addresses the critical challenge of limited labeled Arabic datasets by effectively handling dialectal variations and domain-specific terminology, establishing new methodological benchmarks for Arabic NLP in public service domains.

Abstract: Social media is crucial for gathering public feedback on government services, particularly in the traffic sector. While Aspect-Based Sentiment Analysis (ABSA) offers a means to extract actionable insights from user posts, analyzing Arabic content poses unique challenges. Existing Arabic ABSA approaches heavily rely on supervised learning and manual annotation, limiting scalability. To tackle these challenges, we suggest an integrated framework combining unsupervised BERTopic-based Aspect Category Detection with distance supervision using a fine-tuned CAMeLBERT model for sentiment classification. This is further complemented by transformer-based summarization through a fine-tuned AraBART model. Key contributions of this paper include: (1) the first comprehensive Arabic traffic services dataset containing 461,844 tweets, enabling future research in this previously unexplored domain; (2) a novel unsupervised approach for Arabic ABSA that eliminates the need for large-scale manual annotation, using FastText custom embeddings and BERTopic to achieve superior topic clustering; (3) a pioneering integration of aspect detection, sentiment analysis, and abstractive summarization that provides a complete pipeline for analyzing Arabic traffic service feedback;

check for updates

Academic Editor: Pierluigi Siano

Received: 21 February 2025
Revised: 27 March 2025
Accepted: 5 April 2025
Published: 8 April 2025

Citation: Alotaibi, A.; Nadeem, F. An Unsupervised Integrated Framework for Arabic Aspect-Based Sentiment Analysis and Abstractive Text Summarization of Traffic Services Using Transformer Models. *Smart Cities* **2025**, *8*, 62. https://doi.org/10.3390/smartcities8020062

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

(4) state-of-the-art performance metrics across all tasks, achieving 92% accuracy in ABSA and a ROUGE-L score of 0.79 for summarization, establishing new benchmarks for Arabic NLP in the traffic domain. The framework significantly enhances smart city traffic management by enabling automated processing of citizen feedback, supporting data-driven decision-making, and allowing authorities to monitor public sentiment, identify emerging issues, and allocate resources based on citizen needs, ultimately improving urban mobility and service responsiveness.

Keywords: NLP; deep learning; pre-trained language model; transformers; Arabic aspect-based sentiment analysis; abstractive text summarization

1. Introduction

Social media has emerged as a vital platform for citizens to express their opinions, offering governments and organizations valuable insights into public sentiment on their policies, services, and public issues. These digital interactions not only help identify areas for improvement but also enable governments to understand public needs and adapt accordingly. Sentiment analysis, which examines opinions and emotions in text, operates at three distinct levels: aspect, sentence, and document. While document-level and sentence-level analyses establish the overall emotional orientation of complete texts or individual sentences, they often fail to detect the subtle distinctions of specific opinions within a single text. Within smart city ecosystems, understanding citizen sentiment about traffic services is crucial for effective governance and resource allocation. Traditional approaches to gathering citizen feedback are often costly, time-consuming, and provide only limited insights. ABSA addresses this limitation by identifying sentiment associated with particular aspects, making it highly relevant for real-world applications.

Despite the global significance of Arabic, spoken by over 422 million people, Arabic ABSA faces significant challenges [1]. The intricate nature of Arabic word structure and its diverse dialects pose unique challenges for ABSA and abstractive text summarization (ATS) tasks, resulting in a lack of robust systems capable of handling the language's linguistic intricacies. Conventional emotional tone detection techniques, which depend extensively on systems governed by predetermined rules, often require extensive feature engineering and large amounts of labeled data. Furthermore, the current landscape for aspect category detection (ACD) in Arabic text analysis presents several limitations. Many studies rely on supervised deep learning (DL) approaches, which necessitate the creation of high-quality labeled datasets—a process that demands substantial human effort [2]. While unsupervised approaches exist as an alternative, they often rely on broad linguistic patterns and keywords, leading to inconsistent performance across different domains. Traditional word embedding approaches also involve labor-intensive feature engineering, where researchers must manually design and select relevant features for the model. Additionally, such methods frequently require the use of external resources, such as sentiment lexicons or other linguistic databases. This reliance is time-consuming and limits the scalability and adaptability of these models to new domains or variations in language use [3]. As a result, an urgent requirement exists for a sophisticated framework that can effectively analyze and summarize textual content while preserving its essential meaning and key information.

To tackle these challenges, this study proposes a novel unsupervised approach for ABSA and ATS. The proposed method aims to detect and extract aspect categories from text without relying on labeled data and to determine sentiment polarity associated

with the identified aspects. Our approach minimizes reliance on extensive manually annotated data, requiring only a small set of classified information to evaluate the system's performance. Therefore, our method showcases the potential of a scalable solution for ACD and ABSA tasks across various domains. Our innovative approach leverages clusters of unlabeled posts collected from X (formerly Twitter), focusing specifically on traffic-related services.

Additionally, the study introduces a sophisticated Arabic ATS system specifically designed for traffic services content. By leveraging leading-edge natural language processing models, including mT5 (part of the multilingual T5 family) [4] and AraBART (based on the Bidirectional and Auto-Regressive Transformers (BARTs) architecture) [4], our system effectively identifies and summarizes the key information from Arabic traffic services feedback. This will enable traffic agencies to generate accurate and concise summaries of user sentiment, facilitating a better understanding of public sentiment and service needs. The system's ability to maintain semantic accuracy while condensing large volumes of Arabic text will provide agencies with an efficient tool for processing and analyzing user feedback.

This research makes several significant contributions to the field of Arabic NLP and smart city applications. While previous ABSA studies have focused primarily on domains such as hospitality, retail, and general news, our work is the first to specifically address the traffic services domain in Arabic social media. Moreover, unlike existing approaches that typically rely on supervised learning with extensive manual annotation, our framework employs an innovative unsupervised technique that combines FastText custom embeddings with BERTopic for aspect detection, significantly reducing the annotation burden while maintaining high accuracy. Additionally, our work is unique in its end-to-end integration of aspect detection, sentiment analysis, and abstractive summarization within a single cohesive system, providing not just identification of issues but also a concise explanation of citizen concerns. The CAMeLBERT fine-tuning approach we employ demonstrates that domain adaptation can significantly outperform general-purpose models, achieving a 26.5 percentage point improvement in F1-score compared to the best existing Arabic ABSA methods. Furthermore, our AraBART-based summarization model sets a new state-of-the-art for Arabic abstractive summarization with ROUGE scores substantially exceeding previous benchmarks. In summary, the key contributions of this paper include:

- Developing an unsupervised ACD approach using BERTopic for aspect category detection.
- Comparing static and contextual word embeddings for the Arabic traffic domain to evaluate their impact on ACD performance. To the best of our knowledge, this is the first study to compare these methods using a domain-specific embedding model in Arabic ABSA.
- Training a CAMeLBERT model on an unlabeled dataset to extract aspect categories and sentiment polarity, thereby promoting a data-centric approach. To the best of our knowledge, this approach has not been previously employed.
- Training and fine-tuning AraBART and mT5 models for abstractive summarization of Arabic texts in the traffic services domain.
- Proposing a novel dataset consisting of 56,829 texts with corresponding summaries, advancing research in Arabic ATS.

By combining ABSA and ATS techniques, this study provides actionable insights into public sentiment, enabling decision-makers in the traffic department to identify service gaps, prioritize infrastructure improvements, and respond to emerging traffic issues proactively.

By automating the analysis of Arabic social media content, the framework supports the core smart city principles of citizen engagement, data-driven governance, and responsive service delivery. Furthermore, the framework's unsupervised approach minimizes manual annotation costs while maximizing scalability across different Arabic dialects and traffic-related services, making it particularly valuable for resource-constrained smart city implementations in Arabic-speaking regions.

The developed summarization system effectively condenses user feedback into concise summaries, preserving semantic accuracy and enhancing decision-making efficiency.

The following sections of this manuscript are organized accordingly: Section 2 examines previous work; Section 3 details data collection, preprocessing, and model design; Section 4 discusses findings and performance evaluation; and Section 5 summarizes the contributions and outlines potential research directions.

2. Related Work

This section is divided into three subsections. The first presents an overview of related work on Arabic ABSA in the government services domain. The second subsection discusses transfer learning techniques using pre-trained language models in broad terms and ABSA based on the BERT model in particular. The third subsection highlights research in Arabic ATS.

2.1. Arabic ABSA in Government Services Domain

ABSA is a nuanced form of sentiment analysis focused on recognizing specific aspects within user reviews and determining the associated sentiment for each [5]. In the context of Arabic language reviews, ABSA is a comparatively recent field of research. Before 2015, research in this domain was scarce, as noted by Al-Twairesh et al. [6]. The pioneering work in Arabic ABSA was conducted by Al-Smadi et al. [7], who provided a corpus of book reviews annotated for ABSA, adhering to the guidelines established by the SemEval 2014 task 4 [8]. This work also established a set of baseline models for various ABSA subtasks. The performance metrics across different aspects of their analysis were as follows: Aspect Term Extraction (ATE) achieved an F1-measurement of 23.4%, ACD yielded an F1-measurement of 15.2%, Aspect Term Polarity (ATEP) demonstrated an accuracy of 29.7%, and Aspect Category Polarity (ACP) showed an accuracy of 42.6%. These results laid the foundational benchmarks and highlighted key areas for improvement in Arabic ABSA, paving the way for further research. Subsequently, researchers introduced several benchmark datasets covering various domains to further advance the field [9].

Al-Ayyoub et al. [9] introduced a benchmark dataset for Arabic ABSA on news posts, following SemEval 2014 task 4 guidelines. Their foundational frameworks attained: ATE (F1-score: 39.4%), ATEP (Accuracy: 65.9%), ACD (F1-score: 64.9%), and ACP (Accuracy: 74%). A lexicon-based approach, incorporating additional features like POS tags, N-grams, position, and pruning, improved ATE (F1-score: 51.4%) and ATEP (Accuracy: 67.3%). Areed et al. [10] developed an ABSA dataset for reviews of Dubai government smart applications. Their hybrid rule-based and lexicon-based approach outperformed the baseline, achieving an F1 measurement of 92.5% for aspect extraction (17% improvement) and an accuracy of 95.8% for sentiment classification (6% improvement).

DL techniques in Arabic sentiment analysis, particularly ABSA, are still evolving [5]. Al-Smadi et al. [11] applied LSTM-based models to Hotel datasets [12,13]. For Opinion-Target Expression (OTE), they used the BiLSTM-CRF model with Word2vec and FastText embeddings, while sentiment classification employed an LSTM model with aspect-OTEs as attention mechanisms. The BiLSTM-CRF model with FastText embeddings showed an

improvement of 39% (F1-score: 69.9%). Sentiment classification achieved an accuracy of 82.6%, aligning with the best result of SemEval 2016. Additionally, Al-Smadi et al. [12] proposed two models: BGRU-CNN-CRF for OTE and IAN-BGRU for sentiment polarity classification (SPC). These models were evaluated by Pontiki et al. [13] on hotel review datasets. They achieved a 70.67% F1-score for OTE (39.7% improvement) and 83.98% accuracy for sentiment classification (7.58% improvement), outperforming baseline models and addressing key challenges in Arabic NLP. Recent models leverage pre-trained transformer architectures, significantly improving aspect detection and sentiment analysis accuracy [14]. This progression has reduced reliance on manual linguistic expertise while enhancing ABSA capabilities across various domains.

Arabic ABSA research spans diverse domains, including public services and consumer experiences. Prior studies have focused on analyzing reviews from sectors such as hotels, books [1], news [9], airlines [15], education [16], e-government apps [10], and telecom services [17]. Among these, the hospitality sector, especially hotel reviews, dominates the field, accounting for 42% of studies [18]. However, the transportation and traffic management domain remains largely underexplored, despite its significant impact on daily life. Analyzing tweets about Saudi traffic services using ABSA techniques can provide valuable insights into public sentiment on issues like driving licenses, road conditions, and traffic violations. Our research tackles this gap by concentrating on domain-specific applications of ABSA to provide actionable insights for improving traffic services. We employ transfer learning with pre-trained language models for Arabic ABSA, specifically ACD and APC tasks. Our approach minimizes reliance on complex preprocessing and feature extraction while addressing the scarcity of annotated Arabic ABSA datasets through advanced NLP techniques.

2.2. ARABIC ABSA Using Transfer Learning Models

Transfer learning, introduced by Pratt [19], applies knowledge from one problem to related tasks. In NLP, pre-trained language models like GPT [20], XLNET [21], and BERT [22] have advanced various tasks, including sentiment analysis. These models, pre-trained on extensive unlabeled collections, can be fine-tuned for particular functions with minimal annotated information, addressing resource limitations and reducing computational costs for new model development. BERT is a bidirectional, contextualized language model that generates dynamic word embeddings based on context, unlike static models like Word2Vec and GloVe. It is pre-trained on an extensive collection of unlabeled text, including Wikipedia and books, using two main tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM is referred to forecast masked words in a sentence, whereas NSP determines whether two sentences are related. BERT has accomplished leading-edge outcomes across diverse natural language processing challenges, including sentiment analysis. Several BERT variants have been developed, such as RoBERTa, ALBERT, and DistilBERT, each with specific optimizations or modifications. Multilingual models, such as mBERT (multilingual BERT) [22] and XLM-RoBERTa [23], are trained on multiple languages including Arabic, enabling cross-lingual applications and transfer learning across diverse languages.

Monolingual models, conversely, are specifically trained and optimized for the Arabic language. Examples include Hulmona (ULMfit-based), AraBERT (a BERT-based contextual embedding model), Qarib, and MamBERT (BERT-based). These models have exhibited remarkable proficiency in functions including sentiment analysis, named entity recognition, and question answering. Advancements in both multilingual and monolingual models have notably advanced the performance of NLP tasks in the Arabic language domain.

Abdelgwad et al. [1] explored the application of BERT for APC, using a linear classification layer and testing it on three Arabic datasets: news, book reviews, and hotel review articles. The model achieved promising results across all 3 datasets (89.51% accuracy on hotel reviews, 73.23% on book reviews, and 85.73% on news articles). Their analysis highlighted several findings: the model's robustness against overfitting, the role of fine-tuning BERT, and its ability to handle multiple aspects, conflicting polarities, and negation. These findings demonstrate the significant potential of pre-trained language models like BERT in advancing Arabic ABSA, achieving top-tier results with a relatively simple model architecture.

Chouikhi et al. [24] suggested an integrated framework combining ATE and APC to address the research gap in Arabic NLP. They utilized an Arabic BERT model fine-tuned on the Human Annotation of Arabic Dataset (HAAD). They enhanced the BERT model by adding a Conditional Random Field (CRF) layer, creating an Arabic-BERT-CRF architecture. This integrated framework surpassed earlier cutting-edge methodologies, achieving 95.23% accuracy and an F1 score of 47.63%. The investigation demonstrated the potential of leveraging pre-trained language models for specific ABSA tasks in Arabic, addressing obstacles posed by limited annotated collections. Their approach showed improved performance and robust handling of complex Arabic linguistic phenomena; however, it requires significant computational resources and the need for domain-specific fine-tuning. Key insights from this work include the value of transfer learning, joint modeling of ABSA subtasks, and the efficacy of pre-trained language models for Arabic natural language processing.

Almasri et al. [25] investigated a semi-supervised approach for Arabic ACD employing BERT and a teacher-student model. They combined labeled SemEval 2016 data with unlabeled HARD 2016 data, implementing a noisy student framework with AraBERT v02. This method achieved a micro F1-score of 0.682, emphasizing the need for utilizing unlabeled data in ACD tasks. The study underscores the value of semi-supervised and transfer learning techniques in overcoming challenges posed by limited labeled data in Arabic ABSA.

Ameur et al. [26], addressed the challenge of ACD in Arabic hotel reviews utilizing a multi-category learning methodology with AraBERT. Their method combined Arabic-specific preprocessing, contextual word embedding through AraBERT fine-tuning, and a dynamically weighted loss function to address imbalanced data. This approach achieved leading-edge performance on the Arabic SemEval-2016 dataset, with an F1 measurement of 67.3%. The investigation emphasizes the significance of language-specific preprocessing, the potential of transformer-based models in capturing contextual information, and the effectiveness of addressing class imbalance in multi-label classification tasks.

Fadel et al. [27] introduced novel approaches to Arabic ABSA using multi-task learning and pre-trained language models. Their MTL-AraBERT model simultaneously performed ATE and ACD, attaining leading outcomes on the Arabic SemEval-2016 dataset (F1 measurement of 68.21% for ACD and 80.32% for ATE). Additionally, their BERT-SPC model combined with BiLSTM or BiGRU achieved high accuracy for APC (89.67%) and ACPC (89.36%) tasks. This research highlights the possibility of integrating transfer learning, multi-task learning, and deep neural networks to advance Arabic ABSA, particularly in treating the obstacles presented by restricted annotated data and the complexity of the Arabic language.

Bensoltane and Zaki [3] explored various word representation frameworks for Arabic ACD employing a BiGRU-based methodology. Their study compared word-level, character-level, domain-specific, and contextualized embeddings on the Arabic SemEval-2016 hotel reviews dataset. The proposed BiGRU model with AraBERT embeddings achieved leading

performance with an F1 score of 65.5%, outperforming previous approaches. This work highlights the importance of selecting appropriate word embeddings and the potential of bidirectional recurrent neural networks in capturing contextual information and handling the morphological complexity of Arabic.

Overall, the literature on Arabic ABSA emphasizes the significance of transfer learning and pre-trained language models in enhancing the accuracy and efficiency of aspect-level sentiment analysis tasks. Transfer learning has demonstrated considerable advantages in semantic representation through pre-trained contextual models, as evidenced in recent studies. By harnessing pre-trained language models and external knowledge bases, researchers strive to enhance the efficiency and effectiveness of ABSA systems for Arabic text analysis, addressing the distinctive challenges presented by the Arabic language's complex word structure and regional linguistic differences. Unlike the previous methods, our approach utilizes the BERTopic model and examines various numerical representational frameworks, encompassing word-level, sentence-level, and contextualized term embedding methodologies to further improve ABSA functionalities.

2.3. Arabic ATS Using Transfer Learning Models

Text summarization has become increasingly vital to the rapid expansion of digital information, particularly in addressing the challenges of Arabic language processing. Recent advancements in DL have revolutionized Arabic text summarization, enabling the generation of concise summaries while preserving the core meaning of source texts. Kamal Eddine et al. [28] introduced AraBART, the first end-to-end pre-trained Arabic sequence-to-sequence model for abstractive summarization. AraBART achieved state-of-the-art results, outperforming established models such as Arabic BERT, multilingual mBART25, and mT5. However, its main limitation is its tendency to produce non-factual or ungrammatical output.

Alqahtani and Al-Yahya [29] investigated the effectiveness of AraBART for Arabic ATS employing three distinct datasets: headline (AHS), highlight (WikiLingua), and full summary (XL-Sum) generation. The model, trained on 300K examples, achieved notable results with ROUGE-1 = 55%, ROUGE-2 = 40.15%, ROUGE-L = 54.55%, and BERTScore = 88.06%. Their work established important benchmarks for Arabic abstractive summarization and provided practical guidelines for optimizing transformer models for different summarization tasks in Arabic. However, the absence of comprehensive human evaluation left some uncertainty about the real-world readability and effectiveness of the model's outputs.

Elsaid et al. [4] suggested a comprehensive methodology for Arabic text summarization using mT5 and AraBart transformers. They developed two Arabic datasets, HASD (43,000 texts) and AASD (150,000 texts), along with novel evaluation metrics. Their experimental results demonstrated mT5′s superior performance over AraBart across various ROUGE metrics, achieving scores of 0.76, 0.62, and 0.75 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, on the AASD dataset. However, the proposed Arabic-Rouge metric faced challenges with words missing from Arabic WordNet, and some technical implementation details lacked clarity.

Masri et al. [30] focused on Arabic text summarization for educational content, developing and evaluating multiple transformer-based models on biology textbook data. They created a specialized dataset and compared the performance of four transformer models: mT5, AraBART, AraT5, and mBART50. AraBART achieved the highest expert rating of 8.409, followed by mBART50 at 8.18, while mT5 scored lower at 7.17. In terms of automated metrics, AraBART demonstrated strong performance with ROUGE-1 scores of 0.2462, whereas mT5 showed limited effectiveness with ROUGE scores around 0.06. Their

study is limited by the small dataset size (111 examples) and the gaps between automated metrics and human evaluation scores.

Einieh et al. [31] investigated Arabic ATS by fine-tuning the AraT5 transformer framework on a collection of 267,000 Arabic news articles. AraT5 outperformed sequence-to-sequence models, achieving scores of 0.494, 0.339, and 0.469 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, along with a BLEU score of 0.4224. The study provided valuable empirical evidence for the superiority of transformer-based approaches over traditional sequence-to-sequence architectures in Arabic summarization.

Arabic ATS has seen significant advances with the emergence of transformer-based architectures. Early approaches primarily relied on sequence-to-sequence models, but recent studies demonstrate the superior performance of pre-trained transformer frameworks, for instance, AraBART, AraT5, and mT5. There is a clear progression from traditional DL approaches to more sophisticated transfer learning methods, with particular success in handling the complex morphological structure of Arabic text. Notable works by Eddine et al. [28] with AraBART and Elsaid et al. [4] with mT5 have established new benchmarks in Arabic summarization performance, achieving ROUGE-1 scores above 0.49 for AraBART and 0.76 for mT5. However, several challenges remain, including limited dataset availability, the need for robust evaluation metrics specific to Arabic, and the dominance of news domain data in existing research. Moreover, most studies focus on single-sentence summaries, leaving room for exploration in generating longer, more complex summaries. These gaps present opportunities for future research, particularly in developing more diverse datasets and evaluation methodologies that better capture the nuances of Arabic language summarization. In conclusion, the domain of ATS continues to progress with the implementation of new models and datasets tailored to specific languages such as Arabic.

3. Methodology

Our proposed system addresses key challenges in understanding user feedback by automating three crucial tasks:

1. Aspect Category Detection: This task identifies and classifies the topic or category being discussed in a given text, such as “traffic lights”, “road conditions”, or “driving licenses”, from a predefined set of categories [13]. By accurately categorizing text into relevant topics, this step provides a structured representation of the dataset’s content.
2. Aspect Polarity Classification: After identifying the aspect, this task determines the sentiment polarity linked with each category. Sentiment is categorized as positive, negative, or neutral, depending on the emotional tone expressed in the user feedback. This helps to gauge public opinion and satisfaction levels for specific traffic services.
3. Abstractive Text Summarization: This task generates concise and human-readable summaries of users’ opinions, highlighting the reasons behind their sentiments. The summaries explain why users express positive or negative feedback regarding traffic services, providing actionable insights for decision-makers. An example of this is illustrated in Figure 1.

Our proposed framework addresses key challenges in Arabic ABSA and text summarization for traffic services through an integrated unsupervised approach. The methodology combines three main components into a unified pipeline: Aspect Category Detection (ACD), Aspect Polarity Classification (APC), and Abstractive Text Summarization (ATS). Figure 2 presents the comprehensive workflow of our system, illustrating how these components interact. The pipeline begins with social media data collection and preprocessing tailored

for Arabic text. This processed data then flows through our ACD module, which employs customized FastText embeddings and dimensionality reduction through UMAP before applying HDBSCAN clustering and BERTopic to identify relevant traffic service categories without manual annotation. Next, the identified aspects pass to our APC module, which leverages a fine-tuned CAMeLBERT model to determine sentiment polarity for each aspect. Finally, the ATS module generates concise explanatory summaries using a fine-tuned AraBART transformer model. A key innovation in our approach is the careful integration of these components to handle the linguistic peculiarities of Arabic traffic-related social media content. The framework applies domain adaptation techniques at each stage to maximize performance in this specialized domain. For embeddings, we customize FastText to capture traffic terminology across Arabic dialects. For clustering, we optimize HDBSCAN parameters (min_cluster_size = 512, min_samples = 8) specifically for the distribution patterns in traffic-related discussions. For sentiment analysis, we implement a weighted loss function to address the natural imbalance in sentiment distribution. For summarization, we train against references generated with linguistic domain adaptation.

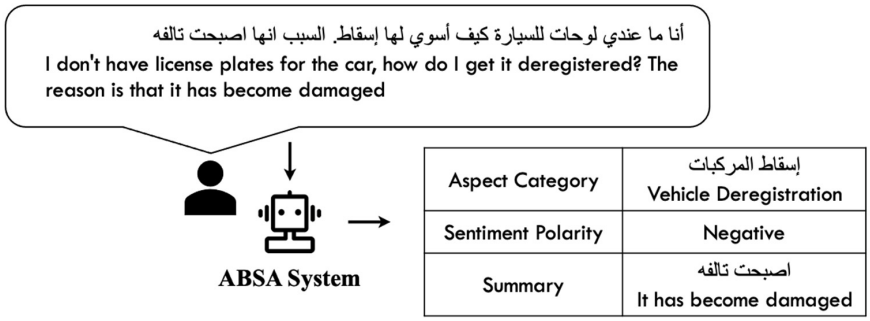


Figure 1. End-to-end analysis of an Arabic traffic services tweet: Integrated aspect category detection, sentiment polarity classification, and abstractive summarization.

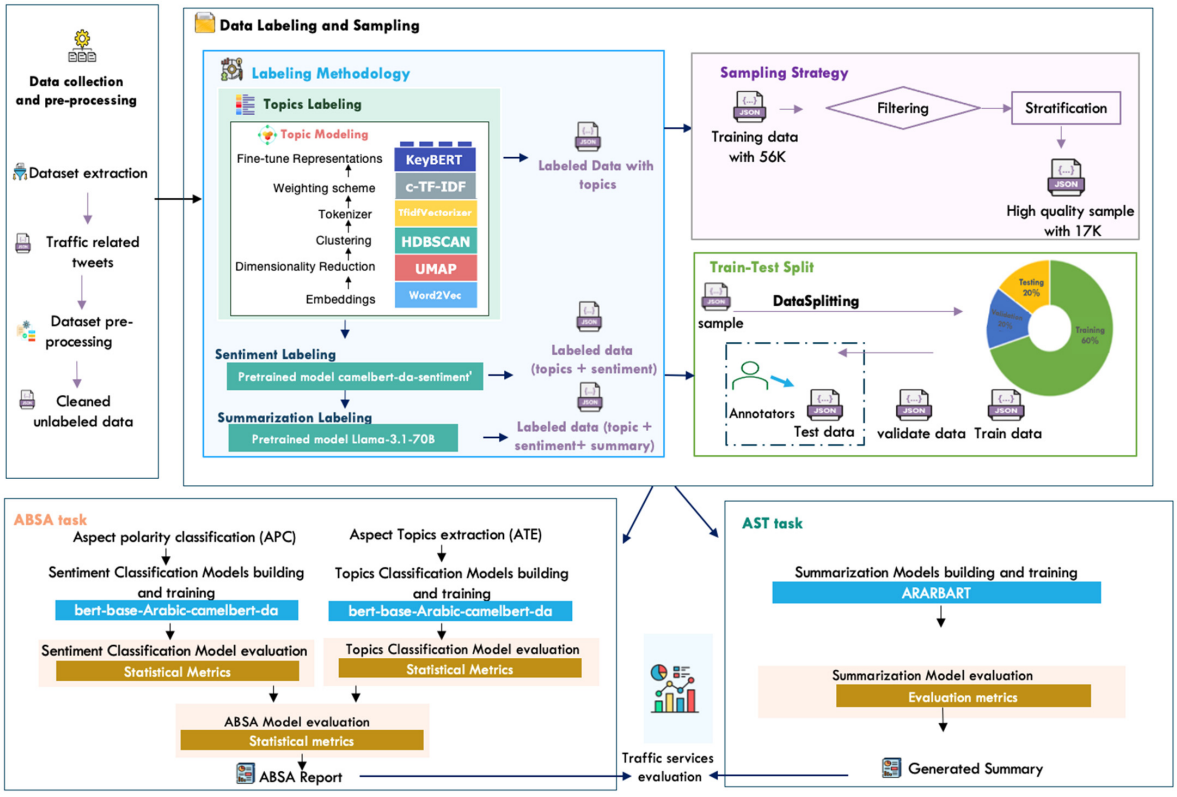


Figure 2. The framework of the current study.

The following sections detail the implementation of each component, beginning with our specialized data collection and preprocessing approach for Arabic social media text.

3.1. Dataset Development

We collected tweets about traffic services from 1 June 2022 to 29 April 2023. The dataset included original tweets, replies, and mentions related to traffic services. To ensure comprehensive coverage, we identified three of the most popular Twitter accounts frequently associated with traffic-related discussions:

1. @eMoroor: The official account of Saudi Traffic, which provides updates and information about traffic services.
2. @MOISaudiArabia: The official account of the Ministry of Interior, which oversees various public services, including traffic-related issues.
3. @Absher. The account is linked to Absher, a leading e-services platform in Saudi Arabia where citizens and residents inquire about traffic services.

Using Twarc2, a Python 3.8 library and command-line tool for archiving Twitter data, we executed search queries to retrieve historical tweets from the specified accounts. The queries captured all mentions directed to the above three accounts within the specified date range. Each tweet was saved as a JSON object returned by the Twitter API and stored in a JSONL (JSON Lines) file format for further processing.

The extracted data from all three accounts was converted into .csv files and then merged into a unified dataset. This consolidated dataset consisted of 461K tweets, including original tweets, replies, and mentions related to traffic services. Table 1 presents a distribution of the number of tweets collected from each account.

Table 1. Number of tweets for each account before the preprocessing step.

From eMoroor	53,493	Mentions_eMoroor	138,415
From Absher	89,055	Mentions_Absher	131,259
From MOISaudiArabia	517	Mentions_MOISaudiArabia	49,105
Total	143,065	Total	318,779
Total = 461,844			

To prepare the dataset of 461,000 tweets for analysis, we performed a comprehensive preprocessing step. We used a Python function, clean_text, which processes the raw tweets by removing extraneous elements and standardizing the text. This function is specifically designed for Arabic content accumulated from social networking platforms, such as X. The preprocessing operations focus on cleaning and structuring the data while addressing the unique linguistic features of Arabic text.

The preprocessing workflow includes the following key points:

1. Removing Hashtags, Mentions, and Links: The function removes hashtags (including any trailing text), user mentions, and hyperlinks using regular expressions. This step ensures that non-informative elements that do not contribute to the semantic content of the tweets are excluded.
2. Applying AraBERT Preprocessing: The preprocessing pipeline utilizes AraBERT's preprocessing function arabert_prep.preprocess(), which incorporates Farasa segmentation as an integral component. Farasa is a widely used tool for Arabic text processing that breaks down words into their smallest meaningful units, known as morphemes. This is particularly important for Arabic due to its complex morphological structure. The AraBERT preprocessing step includes:

Segmenting text into morphemes using Farasa.
Normalizing characters, such as standardizing different forms of the same letter.
Removing diacritics and special characters to simplify the text.

These steps align the dataset's text format with the input requirements of AraBERT, improving model performance.

3. Reversing Preprocessing with AraBERT Unpreprocessing: After segmentation and normalization, the arabert_prep.unpreprocess() function is applied to reverse some of the changes made during preprocessing. This step restores the segmented text to a more human-readable form for downstream tasks. The unpreprocessing process involves:
   - Reinserting spaces removed during preprocessing.
   - Restoring characters and sequences that were standardized.
   - Reintroducing diacritics or original character variations where necessary.

   By combining these preprocessing steps, the raw text is transformed into a clean and standardized format suitable for analysis while preserving its linguistic integrity. This methodology guarantees the dataset is compatible with the requirements of pre-trained Arabic language models, including AraBERT, used in this study.
   Table 2 provides an example of an Arabic sentence and demonstrates its processing through the AraBERT preprocess and unpreprocess functions.

Table 2. Examples of preprocess and unpreprocess functions for Arabic text.

Original text (Arabic)	@MOISaudiArabia‘السلام عليكم**\r\n** عندي رخصة موقتة وتم تسديد رسوم التجديد وعند محاولة تجديد الرخصة بالموقع لا تفتح ايقونة تحديد الرخصة. كيف يمكنني تجديد الرخصة؟’@eMoroor
English Translation	@MOISaudiArabia I have a temporary license, and I've paid the renewal fees. When I try to renew my license on the website, the license selection icon does not work. How can I renew my license? @eMoroor
After Farasa segmentation (using the arabert_prep.preprocess())	ال+ سلام علي +كم عند +ي رخص +ة موقت +ة و+ تم تسديد رسوم ال+ تجديد و+ عند محاول +ة تجديد ال+ رخص +ة ب+ ال+ موقع لا تفتح ايقون +ة تحديد ال+ رخص +ة . كيف يمكنني تجديد ال+ رخص +ة
After model processing and arabert_prep.unpreprocess()	السلام عليكم عندي رخصة موقتة وتم تسديد رسوم التجديد وعند محاولة تجديد الرخصة بالموقع لا تفتح ايقونة تحديد الرخصة. كيف يمكنني تجديد الرخصة

These functions standardize the text for NLP tasks while maintaining its core content and structure. The preprocessed and unpreprocessed text aligns with the original message, making it suitable for further NLP tasks or analysis.

4. Further Cleaning by clean_text(): After AraBERT preprocessing and unpreprocessing, the clean_text() function further refines the text by removing or replacing specific strings, including:
   - Removing placeholders such as "[مستخدم]" (user)"[رابط]",]" (link), "RT" (retweet), and colons.
   - Replacing "[بريد]" (email) with an empty string.
   - Replacing underscores, "#" symbols, and new line characters ("\n") with spaces.
   - Replacing multiple consecutive whitespace characters with a single space.
5. Whitespace Trimming: The strip() function (from the Python library) is applied to remove leading and trailing whitespace from the text. This ensures the cleaned text string is free from unnecessary blank spaces or formatting artifacts.
6. Length Analysis for Topic Modeling: Once the tweets are cleaned, their lengths are calculated to prepare for topic modeling. Under X's previous policy, tweets

were limited to 280 characters. Extremely short tweets (for example, fewer than 50 characters) are challenging for topic modeling due to insufficient context. Our analysis of the dataset revealed that most tweets range from 50 to 120 characters, providing adequate content for further processing.

7. Tokenization Process: Tokenization is performed on the cleaned tweets to break the text into individual words. Using regular expressions, only words of at least two characters are retained. This ensures meaningful tokens for analysis while discarding irrelevant or overly short strings. Table 3 illustrates a sample of the tokenization process.

Table 3. Sample of Tokenization process of Traffic tweet.

Original text	استفسار لو تكرمتو لدي اثر اصتدام بالسيارة خفيف ولم يتم صيانته لعدم القدرة للوصول للمتسبب من قبل المرور, وبعد ذلك جاني حادث اصدام بنفس موقع الحادث الأول من قبل شخص ولدي بياناته, ماهو الإجراء في هذه الحالة هل يتم المتابعه من قبل نجم والتقدير مره أخرى؟ ام ماهو الإجراء المناسب؟!
English Translation	I have a minor collision impact on my car that was not repaired because I could not reach the person who caused it through traffic authorities. After that, I had another collision accident at the same location as the first accident by a person whose information I have. What is the procedure in this case? Should I follow up with Najm (Saudi insurance company) and get another estimate? Or what is the appropriate procedure?!
After tokenization process	استفسار', 'لو', 'تكرمتو', 'لدي', 'اثر', 'اصتدام', 'بالسيارة', 'خفيف', 'ولم', 'يتم', 'صيانته', 'لعدم', 'القدرة', 'للوصول', 'للمتسبب', 'من', 'قبل', 'المرور', 'وبعد', 'ذلك', 'جاني', 'حادث', 'اصدام', 'بنفس', 'موقع', 'الحادث', 'الأول', 'من', 'قبل', 'شخص', 'ولدي', 'بياناته', 'ماهو', 'الإجراء', 'في', 'هذه', 'الحالة', 'هل', 'يتم', 'المتابعه', 'من', 'قبل', 'نجم', 'والتقدير', 'مره', 'أخرى', 'ام', 'ماهو', 'الإجراء', 'المناسب'

Finally, the dataset contains a total of 1,150,260 tokens, with 56,997 unique tokens. The large number of tokens demonstrates that the dataset is rich in linguistic content, ensuring an effective representation of the language's complex structure. After performing comprehensive preprocessing steps on our dataset, the final clean dataset consisted of 56,829 records, which were then labeled into several categories for topics and sentiments. For sentiment classification, texts were labeled as positive, negative, or neutral using pre-trained sentiment analysis models. To categorize the data into topics, we employed BERTopic, a topic modeling technique that generates high-quality topic labels based on semantic similarity and clustering.

3.2. Model Development

3.2.1. Topic Modeling

1. Generating Embeddings

Word embeddings, a technique for representing words as numerical arrays in a quantitative vector space, capture semantic associations by positioning words with comparable meanings in greater proximity [32]. These embeddings can be broadly categorized into static and contextualized word embeddings.

Static embeddings assign fixed representations to words regardless of their context. Examples include Word2Vec and FastText. Word2Vec operates using two fundamental architectures: Continuous Bag of Words (CBOWs) and Skip-gram. CBOW forecasts a central term derived from adjacent words within a defined range, whereas Skip-gram anticipates the surrounding context utilizing a central term [3]. Skip-gram demonstrates particular effectiveness for processing uncommon terminology, as it assigns them higher weight compared to CBOW. However, a key limitation of static word embeddings is their inability to handle polysemy, as a single word always has the same representation, regardless of its context.

Contextualized word embeddings address this limitation by generating representations that vary depending on the surrounding context. A prominent example is the BERT (Bidirectional Encoder Representations from Transformers), which uses a transformer-based architecture to process text directionally, considering both preceding and following words for a comprehensive understanding of the context. During unsupervised pre-training, BERT implements methodologies such as MLM to attain cutting-edge results across numerous natural language processing challenges, including question answering, sentiment analysis, and named entity recognition.

We conducted a comprehensive comparison of multiple embedding approaches to identify the optimal representation for Arabic traffic-related content. Our evaluation included static embeddings (Word2Vec and FastText), contextual embeddings (AraBERT), and sentence-level embeddings (DistilUSE). For each model, we assessed topic coherence and clustering quality.

- Word-level representations: We used FastText, an extension of Word2Vec, trained by Facebook utilizing Common Crawl and Wikipedia information employing CBOW architecture [33]. FastText enhances the original Word2Vec model by incorporating subword information, allowing it to generate better representations for rare words and even out-of-vocabulary terms. Additionally, we customized FastText embedding for our task and made it publicly available [34].
- Sentence-level representations: For sentence embeddings, we utilized the DistilUSE (Distilled Universal Sentence Encoder) model [35], specifically the variant "sentence-transformers/distiluse-base-multilingual-cased-v1". This model, part of the sentence-transformers framework, generates robust multilingual sentence-level embeddings.
- Contextualized word embeddings: We employed AraBERT version 2, a pre-trained language model specifically designed for Arabic [36]. This model leverages the MLM approach to generate contextualized representations. Specifically, we utilized the "aubmindlab/bert-base-arabertv2" variant of AraBERT. Sentence representations were computed by averaging the contextualized representations of individual tokens within each sentence.
- Domain-specific Contextualized word embeddings: We fine-tuned AraBERT version 2 to adapt the model's parameters for generating embeddings tailored to traffic services. The model was loaded using the Sentence-Transformers package. This process enabled us to tailor the general-purpose Arabic language model to our specific domain, potentially improving its performance on traffic-related tasks. Our model is available to the public [37]. Again, sentence representation was computed by averaging the contextualized representations of individual tokens.

2. Dimensionality Reduction

Dimensionality reduction represents an essential methodology in data analysis and machine learning that reduces the number of attributes (e.g., irrelevant) in a collection while maintaining the most crucial elements. This process minimizes data storage requirements and decreases processing durations. By reducing the number of dimensions, dimensionality reduction also improves the efficiency of clustering algorithms. This allows text mining techniques to operate on data with fewer terms, thereby enhancing overall performance [38]. In this study, we used the UMAP (Uniform Manifold Approximation and Projection) algorithm to decrease representation dimensionality. UMAP is a highly regarded dimensionality reduction method, known for its robust mathematical principles. Its proficiency in managing non-linear relationships and its ability to retain the majority of information within a smaller number of dimensions have made it a popular choice. The widespread adoption and success of UMAP across

various scientific fields demonstrate its exceptional capabilities and versatility as an algorithm [38]. In our study, we used the FastText model, reducing the dimensionality from 100 to 8. Through hyperparameter tuning, we found that 8 dimensions yielded the best results for enhancing topic modeling.

3. Clustering

Clustering constitutes an unsupervised computational learning methodology that organizes similar points into cohesive groups, called clusters. In the context of topic modeling, clustering facilitates the organization of topics into distinct categories, enabling more effective classification within each cluster. In this study, we used the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm for document clustering. HDBSCAN is a hierarchical clustering algorithm known for its ability to handle datasets with varying densities and its resistance to interference and anomalies. One key feature of HDBSCAN is that it does not necessitate predetermined cluster quantities, a limitation often encountered with algorithms such as K-Means. Unlike K-means, HDBSCAN does not assume spherical clusters and automatically determines the optimal number of clusters. It also produces separate clusters for outliers, reducing noise in the clustering process [39]. This makes HDBSCAN especially valuable in topic modeling contexts, where the number of topics is unknown and topics may vary in densities or sizes in the feature space.

In our study, we applied HDBSCAN to cluster the 8-dimensional embeddings generated by FastText and reduced using UMAP. HDBSCAN relies on two main parameters that influence the cluster selection: min_clustersize and min_samples. We set min_clustersize to 512, meaning that any cluster with fewer than 512 elements will be considered as noise. The min_samples parameter was set to 8, controlling the creation of a mutual reachability graph used for clustering.

4. Vectorization

The next phase in the process involved data vectorization. Utilizing the BERTopic technique, we represented cluster topics through the Term Frequency-Inverse Document Frequency (TF-IDF) method. TF-IDF calculates the importance of words within individual documents relative to the entire corpus. We employed the TfidfVectorizer with specific parameters to optimize topic representation.

Our key settings included:

- Emphasizing phrases consisting of 2 to 5 words.
- Excluding infrequent terms (those appearing in less than 10% of documents).
- Removing very common terms (those appearing in over 60% of documents).
- Omitting stop words.
- Retaining all features without imposing vocabulary limitations.

This methodology enabled us to comprehend the significance of each word in each sentence across the topics, improving the quality of topic representation.

5. Topic Modeling using BERTopic

Topic modeling is an unsupervised learning methodology that requires no annotated data or pre-trained models for topic understanding. In this study, we employed *BERTopic*, a topic modeling framework that uses the HDBSCAN algorithm to automatically identify the ideal number of topics. BERTopic was selected based on its superior performance compared to traditional topic modeling techniques in several studies. Abuzayed and Al-Khalifa [40] showed that BERTopic outperformed both LDA and NMF across multiple datasets, achieving higher NPMI (Normalized Pointwise Mutual Information) and topic diversity scores. Their experimental results demonstrated that BERTopic utilizing Arabic

monolingual pre-trained models exhibited particularly strong performance. Similarly, Aouichaty et al. [41] emphasized BERTopic's effectiveness for legal Arabic text, where it provided better contextual representation than conventional approaches. Additionally, Alhaj et al. [42] successfully applied BERTopic to improve cognitive distortion classification in Arabic Twitter content. Their study showed that employing latent topic distribution acquired from BERTopic significantly enhanced classifier performance in distinguishing between different cognitive distortion categories. Uthirapathy and Sandanam [43] further confirmed BERTopic's effectiveness when combined with BERT-based classification for climate change Twitter data, achieving a precision of 91.35%, recall of 89.65%, and accuracy of 93.50%. These studies collectively demonstrate that BERTopic, with its ability to leverage contextual embeddings from transformer models, consistently outperforms traditional topic modeling approaches like LDA and NMF, particularly for tasks involving shorter texts and complex semantic relationships.

Based on BERTopic, the HDBSCAN identified 16 unique topics by clustering the data. The framework was configured to eliminate Arabic stop words and filter out terms with document frequencies outside the range of 10 to 60, thereby reducing the size of the resulting sparse c-TF-IDF matrix. The KeyBERTInspired word representation model was used to identify a list of topics, with each topic characterized by the most relevant words, helping to understand the major themes within a document collection. This unsupervised approach is designed to be both simple and effective, identifying key terms by comparing cosine similarity between word and document embeddings [44].

The BERTopic framework consists of four main phases:

1. Document Clustering: Documents were first clustered into topics using embeddings and clustering algorithms.
2. Topic Documents Creation: All documents within each topic were combined into a single, large "topic document".
3. Vectorization and c-TF-IDF Calculation: The vectorizer model was applied to these topic documents, and c-TF-IDF (Class-based TF-IDF) was calculated. c-TF-IDF calculates the significance of terms within topics by considering Term Frequency (TF), which quantifies how regularly a word occurs in a topic document, and Inverse Document Frequency (IDF), which calculates how uncommon a word is across various topics. This approach ensures that the words most unique to each topic are highlighted.
4. Keyword Extraction: The KeyBERTInspired process was used to extract keywords by generating embeddings for all documents in the corpus. Topic embeddings were generated by averaging the document embeddings within each cluster. Representative keywords were then identified by comparing word embeddings to topic embeddings. To obtain probable keywords for each document, we used TF-IDF and a bag-of-words technique. These candidate keywords were embedded and compared to the document embeddings. Keywords were ranked according to their similarity to the document embeddings, and the top-scoring keywords were finally chosen to represent each document or topic, ensuring that the selection accurately conveyed the semantic meaning of the material. The similarity score was computed using the following equation:

$$\text{Score} = \text{COS}(D, K) \tag{1}$$

where D is the document embedding vector and K is the keyword embedding vector. Cosine similarity ranges from -1 to 1, where 1 signifies complete similarity, 0 indicates no similarity, and -1 represents perfect dissimilarity.

Our results identified the 10 most relevant words for each topic, based on their importance in the context of the topic. To optimize memory usage in UMAP, we

enabled the low_memory parameter. We also set calculate_probabilities to True to determine the probability of document-topic associations and enabled the verbose parameter to monitor model progression stages. For topic determination, we utilized the automatic mode by setting the number of topics parameter to 'auto'.

6. Topics Reduction

When using HDBSCAN, several outliers are generated that do not fall into any of the generated topics and are classified as -1. In our case, the number of outliers was high, necessitating a reduction in the documents classified as outliers. BERTopic handles outliers through its main component, the reduce_outliers algorithm, which employs an approximate_distribution strategy. This algorithm optimizes processing large datasets by randomly selecting document samples and calculating their topic distributions. It identifies documents with unusual topic distributions and moderates their influence, thereby creating more robust and representative topic models. This approach effectively reduces the impact of anomalous documents on the topic modeling process.

3.2.2. Sentiment Labeling

To classify the sentiments in our dataset, we categorized texts into positive, negative, and neutral sentiments using pre-trained models. Several benchmark models have been developed for Arabic sentiment classification, trained on a large corpus to detect subtle linguistic variations. Among these, AraBERT and MARBERT have demonstrated strong performance in understanding the distinctive linguistic characteristics of Arabic, while CAMeLBERT, another pre-trained model optimized for Arabic, is particularly proficient at handling various Arabic dialects. Additionally, models like XLM-RoBERTa and AraGPT have shown good performance in understanding the Arabic context.

Among these options, MARBERT and CAMeLBERT stand out for their superior performance in Arabic sentiment classification tasks. For this study, we selected CAMeLBERT, a variant of BERT specifically designed for Arabic language processing. CAMeLBERT was pre-trained on diverse sizes of Arabic texts and variants, encompassing Modern Standard Arabic (MSA), Classical Arabic (CA), and Dialectal Arabic (DA).

Inoue et al. [45] developed and released three distinct CAMeLBERT model types tailored to these variants: an MSA model, a DA model, and a CA model. Additionally, they introduced a combined model pre-trained on a mix of all three variants, offering flexibility for different Arabic language processing tasks. Table 4 provides the configuration details of CAMeLBERT models [45].

Table 4. Configuration of CAMELBERT models [45].

Model	Variants	Size	No. of Words	Tokens	Vocab	No. of Steps
CAMeLBERT-MSA	MSA	107 GB	12.6 B	WP	30 K	1 M
CAMeLBERT-DA	DA	54 GB	5.8 B	WP	30 K	1 M
CAMeLBERT-CA	CA	6 GB	847 M	WP	30 K	1 M
CAMeLBERT-Mix	MSA/DA/CA	167 GB	17.3 B	WP	30 K	1 M

To classify tweets related to traffic services, we employed distant supervision, as the pre-trained model was initially trained for sentiment classification in a different domain (other than traffic services on X). Specifically, we utilized the CAMeLBERT-DA-sentiment (https://huggingface.co/CAMeL-Lab/bert-base-arabic-camelbert-da-sentiment, accessed on 3 April 2025) model, pre-trained on DA datasets to categorize tweets into positive, negative, and neutral classes.

3.2.3. Abstractive Text Summarization

For the ATS task, we generated sentiment summaries for each tweet, identifying whether the feedback contained positive or negative sentiments about traffic services. A sample of 17,338 tweets was used to train and validate the model. Initially, we annotated the dataset with a summary of feedback about traffic services using GPT-4o-mini, a large language model (LLM) developed by OpenAI's.

GPT-4o-mini is the newer variant of OpenAI's GPT model series. This model was released in 2024 and was designed to compete with other advanced LLMs in tasks like NLP, text generation, and sentiment analysis. We utilized this model due to its performance in benchmarks compared to other AI models like GPT-4, Llama, and Claude 3.5 Sonnet.

The system prompt is designed to instruct the GPT-4o-mini in examining the Arabic tweets related to feedback about traffic services in KSA. The prompt specifies that the model should:

- Summarize the user's feedback or complaints in 2–5 words formatted within a JSON object with a single 'summary' key.
- Return 'None' for tweets that are questions or unrelated to the task.
- Do not provide any additional explanations or feedback and adhere strictly to the JSON format.
- Follow the sample tweet and response format provided as examples.

This approach ensures consistent and focused analysis of service-related feedback. We implemented this approach for both training and validation sets.

3.2.4. Sampling and Splitting Strategies

After the data labeling process, we adopted a targeted sampling approach for the inclusion of high-quality labeled data in our analysis. We utilized conditional filtering techniques to extract data points that met predefined criteria aligned with the research objectives. The process began by filtering tweets with a sentiment confidence score of 85% or higher, ensuring the reliability of sentiment annotations. Subsequently, we chose tweets for which the model demonstrated a topic probability of 65% or higher, indicating reasonable confidence in the assigned topic. Following that, we eliminated all tweets that belonged to outlier categories or were unclassified to maintain the dataset's consistency and relevance. As a result of this rigorous sampling process, we curated a dataset containing 17,338 records, which was used for training and evaluating the models.

The next process is the partitioning of the dataset into three distinct sets: training, validation, and test sets. The training set was used to train the model to recognize patterns and relationships within the data, while the validation set was employed to fine-tune hyperparameters and mitigate overfitting. The test set, kept completely separate from the other subsets, served as the ultimate benchmark to evaluate the framework's effectiveness. To maintain data integrity and prevent any information leakage, the test data had to be entirely distinct from both the training and validation datasets.

We employed stratification as a splitting method to preserve class distribution across all subsets. This approach was particularly beneficial given the imbalanced nature of the dataset, where some classes were significantly underrepresented. By employing stratified random sampling, we ensured that the model was trained, validated, and evaluated using samples that accurately represented the data distribution in the original dataset. This approach led to a more accurate and robust model evaluation.

We split the dataset such that 60% (10,402 tweets) was allocated for the training set, while the remaining 40% was divided equally between the validation (3465 tweets) and test (3466 tweets) sets. Finally, to ensure the quality and reliability of our evaluations, the test

set was manually annotated for sentiment, topic, and summary annotations. This manual annotation process is explained below.

To ensure accurate sentiment analysis of traffic-related tweets, we implemented a systematic manual labeling process. This manual annotation of a test set of 3466 tweets aimed to enhance the precision of the transformer model during the training and testing phases. The tweets were classified into three emotional groups: positive (1), neutral (0), and negative (−1).

The annotation process began with the careful selection of three native Arabic-speaking annotators from the traffic department, chosen for their comprehensive knowledge of services and policies to ensure high-quality data labeling. The annotators underwent initial training sessions that included example tweets, followed by practice annotations on a small subset of data. Challenging cases were discussed collectively to ensure consistent interpretation and alignment among the annotators.

Annotators followed strict guidelines to classify tweets into sentiment categories.

- Positive sentiment (1): Assigned to tweets expressing agreement, excitement, or positive phrases highlighting the advantages of traffic services.
- Negative sentiment (−1): Assigned to tweets expressing disagreement, criticism, pessimism, negative phrases, or complaints about shortcomings or challenges in traffic services.
- Neutral sentiment (0): Assigned to tweets lacking clear opinions, containing incomplete information, or asking questions without emotional content.

For tweets with mixed sentiments, annotators were instructed to select the predominant one, defaulting to neutral when no clear emotion was evident.

To categorize topics, we leveraged BERTopic for initial topic modeling, which identified 16 distinct themes within our dataset. These topics were used as annotation categories for manually labeling the test data. During the annotation process, human annotators reviewed each tweet and assigned it to the most relevant topic category from the 16 identified themes. This manual annotation approach ensured that the annotations accurately captured contextual nuances while maintaining consistency with the topic modeling results.

For sentiment-based summarization, we annotated tweets to capture feedback on traffic services, specifically identifying issues or aspects that influenced user satisfaction. This annotation provided a reliable ground truth baseline, mitigating potential biases from automated labeling, and ensured accurate assessment of model generalization through human-verified labels.

The annotation process was conducted iteratively over three rounds, with the 3466 tweets being labeled independently by each annotator to avoid mutual influence. Each round included independent annotations, a thorough review of labeled data, quality control checks, and consistency verification. Labels were modified where necessary to improve accuracy. Throughout the process, regular reviews were conducted to ensure adherence to guidelines and maintain annotation consistency. This systematic approach to manual annotation ensured high-quality training data for our transformer model while maintaining consistency and reliability in the labeling process.

3.3. Aspect-Based Sentiment Analysis Task

ABSA typically comprises three key tasks [13]: ACD, OTE, and APC. ACD identifies the discussed aspect category, such as “traffic lights”, in the sentence “The new traffic lights are efficient”. OTE identifies the particular target of the opinion, while APC identifies the sentiment polarity of each aspect. This paper focuses on ACD and APC due to their crucial role in analyzing aspect-specific sentiments and their application in diverse industries, including e-commerce, hospitality, and public services.

In this research, we developed an ABSA system to analyze traffic services by combining ACD and APC using transformer models. The chosen models were BERT-based models developed by the Computational Approaches to Modeling Language Lab (CAMeL Lab), specifically trained on various Arabic dialects. This model is available through the Hugging Face Transformers library (https://huggingface.co/CAMeL-Lab/bert-base-arabic-camelbert-da, accessed on 3 April 2025) as "bert-base-arabic-camelbert-da". We carefully considered several benchmark language models for Arabic sentiment classification, such as AraBERT, MARBERT, CAMeLBERT, XLM-RoBERTa, and AraGPT, and their potential applications. We ultimately selected CAMeLBERT for its superior performance in handling Arabic dialects and its demonstrated effectiveness with diverse Arabic text variants such as Modern Standard Arabic, Classical Arabic, and Dialectal Arabic, as validated by the study [45]. The model's effectiveness is further substantiated by its impressive performance metrics across multiple benchmark datasets: it achieved an F1-score of 76.9% on ASTD, 93.0% on ArSAS, and 72.1% on the SemEval dataset. Additionally, CAMeLBERT was selected for its superior performance compared to general-purpose multilingual models and its pre-training on DA, making it well-suited for user-generated content.

We fine-tuned this pre-trained model on our specific traffic services dataset, which included 17,338 labeled records. Due to class imbalance in the dataset, we employed a weighted Cross-Entropy loss function, allocating greater importance to uncommon categories and reduced significance to prevalent categories. With these weights, the loss became higher for misclassifications of less frequent classes, thus encouraging the model to better learn from these classes during training.

We fine-tuned CAMeLBERT separately for sentiment analysis and topic modeling, tailoring the hyperparameters for each task. For the Sentiment Classification, we used a learning rate of 1×10^{-5}, a train batch size of 32, and an evaluation batch size of 64. The framework was developed over 5 training cycles with a weight decay of 1×10^{-3} for L2 regularization. We implemented a learning rate warmup strategy with a warmup ratio of 0.05, and the framework's effectiveness was assessed after each epoch.

For the topic classification, we employed a slightly higher learning rate of 2×10^{-5}, while maintaining the same batch size of 64 for both training and evaluation. The topic classifier was trained for 10 epochs with a weight decay of 1×10^{-3} and a warmup ratio of 0.05. The evaluation strategy for both models was set to occur at the end of each epoch. By tailoring the hyperparameters for each task, we aimed to optimize the model's performance for both sentiment and topic classifications in Arabic traffic services-related content. This approach allowed us to account for the specific characteristics and challenges of each task while leveraging the strengths of the CAMeLBERT model.

After fine-tuning, we combined the topic and sentiment classifiers into a unified pipeline to form an end-to-end ABSA system. The system incorporated 16 aspect categories (topics) and 3 sentiment labels (positive, neutral, and negative). The input tweets were processed sequentially through both topic and sentiment models.

We evaluated the system using the gold-labeled test split to assess the system's overall performance. After validation, the ABSA system was applied to the entire dataset of 56,829 tweets, generating a comprehensive ABSA report. This pipeline demonstrated the potential of CAMeLBERT for effectively analyzing sentiment and topics in Arabic traffic-related content, providing actionable insights into user feedback.

3.4. Abstractive Text Summarization Task

ATS encompasses creating a condensed version of text while maintaining its fundamental significance in natural language [28]. For this study, we fine-tuned two summarization models, moussaKam/AraBART (https://huggingface.co/moussaKam/AraBART, ac-

cessed on 3 April 2025) and Google/mT5-small (https://huggingface.co/google/mt5-small, accessed on 3 April 2025), to summarize users' feedback regarding traffic services.

AraBART is the first sequence-to-sequence pretrained model, which has 139M parameters [4]. This BART-based model has been developed for various languages, including Arabic [28], and can be fine-tuned for diverse language generation and understanding functions. AraBART attains superior results on numerous abstractive summarization datasets, surpassing robust benchmarks including Arabic BERT-based models, multilingual mBART, and mT5 models [28]. With its bidirectional encoder and auto-regressive decoder, AraBART is particularly suitable for summarization tasks. The moussaKam/AraBART variant is designed for the Arabic language and fine-tuned for Arabic text summarization.

Google/mT5-small, a part of the T5 (Text-to-Text Transfer Transformer) family, is a multilingual model specially developed for Arabic linguistic operations and trained on a Common Crawl-based collection encompassing 101 languages, including Arabic [4]. Developed by Google Research, mT5 is designed for multiple NLP tasks, including summarization. Among its available variants, we selected the "small" version due to its fewer parameters compared to the base or large versions, balancing performance with computational efficiency.

Both models were fine-tuned to generate summaries in Arabic, focusing on user feedback related to traffic services. The fine-tuning process used a learning rate of 5×10^{-5}, training, and an evaluation batch size of 32 for Moussa's AraBART and 16 for mT5. The model was trained for 5 epochs for Moussa's AraBART and 40 epochs for mT5. Both models were trained with a weight decay of 5×10^{-5} for L2 regularization. We implemented a learning rate warmup strategy with a warmup ratio of 0.05, and the model's performance was assessed at the end of each epoch. For optimization, we utilized the Adam optimization algorithm with beta parameters of 0.9 and 0.999 and an epsilon value of 1×10^{-8}.

Model performance was evaluated at the end of each epoch, allowing adjustments to improve summarization quality. By fine-tuning AraBART and mT5 on our specific dataset, the models were adapted to generate summaries highlighting the reasons behind positive and negative sentiments toward traffic services across various topics. The summaries can provide more nuanced insights into user feedback, offering valuable information for decision-makers.

3.5. Evaluation Criteria

All models of ABSA were evaluated using multiple performance metrics. These included confusion matrices along with Accuracy, Precision, Recall, and F1-scores. We evaluated the effectiveness of the frameworks using both the developed and held-out test sets. To tackle the issue of category imbalance, a weighted loss function was utilized. This function assigns different weights to each class based on the sample size, ensuring the model does not favor classes with more instances.

For summarization models, the effectiveness was measured using Recall-Oriented Understudy for Gisting Evaluation (ROUGE) scores. These metrics evaluate the relevance and information retention in the generated summaries compared to reference summaries. The specific ROUGE evaluation measurements employed were:

$$ROUGE-1 = \frac{\sum unigrams\ in\ reference a^{Count} match^{(Unigram)}}{\sum unigrams\ in\ reference a^{Count\ (unigram)}} \quad (2)$$

$$ROUGE-2 = \frac{\sum bigrams\ in\ reference a^{Count} match^{(bigram)}}{\sum bigrams\ in\ reference a^{Count\ (bigram)}} \quad (3)$$

$$ROUGE-L = \frac{Length\ of\ Longest\ Common\ Subsequence}{Length\ of\ Reference\ Summary} \quad (4)$$

ROUGE-1 assesses the correspondence of individual words (unigrams) between a produced and reference summary. This helps in assessing content capture. ROUGE-2 extends this to pairs of consecutive words (bigrams) to capture language structure and coherence. ROUGE-L evaluates summaries by finding the longest matching word sequences in the same order regardless of proximity, capturing structural similarity and key themes. ROUGE-L helps us understand how well a summary preserves the core flow and organization of reference information.

3.6. Computational Resources and Training Times

To ensure reproducibility and facilitate real-world deployment considerations, we provide detailed information about the computational resources and training times required for our methodology.

3.6.1. Sentiment Classification

The sentiment classifier was trained on an NVIDIA RTX A4000 GPU (16 GiB VRAM). Each training run consumed approximately 7 min for 5 epochs. For hyperparameter optimization, we conducted 30 experimental runs, resulting in a total training time of approximately 210 min. The sentiment labeling process utilized a pre-trained camel-bert-sentiment model, requiring only 2–3 min for the entire dataset.

3.6.2. Topic Classification

Our topic classification pipeline consisted of three distinct phases:

1. Masked Language Model Training: To obtain an embedding model based on araBERT, we utilized high-performance GPU hardware. Each training iteration required approximately 40 min, and we conducted 8 experimental runs for hyperparameter tuning, totaling 320 min of computation time.
2. BERTopic Labeling: Topic labeling was performed on a mid-range GPU with 4 GiB VRAM. Each run required approximately 4 min. We extensively explored the parameter space with 268 experimental configurations (including varying cluster numbers), resulting in 1072 min of processing time.
3. Topic Classifier Training: The topic classifier was trained on our primary GPU for 10 epochs per run. Each training iteration required approximately 15 min, and we conducted 13 experimental runs for hyperparameter tuning, totaling 195 min.

3.6.3. Text Summarization

Our text summarization model utilized the same hardware as the sentiment classifier. Each training run required approximately 10 min for 5 epochs. We conducted 11 experimental runs for hyperparameter optimization, resulting in a total training time of 110 min.

3.6.4. Deployment Considerations

While our experimental evaluations were conducted on consumer-grade hardware (primarily RTX A4000 and RTX 3050 Ti GPUs), production deployment would require scaling computational resources based on anticipated user load. For real-world applications, we recommend deploying on enterprise-grade infrastructure with multiple high-performance GPUs, sufficient system memory, and optimized inference configurations to maintain acceptable response times.

4. Results and Discussion

4.1. Comparison of Embedding Models

The performance of embedding models was analyzed across three key dimensions: topic coverage, outlier detection, and reduced outliers, as illustrated in Figure 3.



Figure 3. Performance comparison of five embedding models.

Figure 3 illustrates the comparison of five embedding models based on different factors.

For the *topic coverage* capabilities of each embedding model, the FastText_Custom exhibited the highest topic diversity (16 topics), closely followed by SBERT_ARABERTv2_CUSTOM and SBERT_DISTIL_USE (14 and 15 topics, respectively). Conversely, the SBERT_ARABERTv2 model showed more focused topic classification, reflecting a trade-off between topic breadth and specificity. This trade-off underscores the diverse capabilities of embedding models in handling content granularity.

For the number of outliers detected by each model before reduction, SBERT_ARABERTv2 identified the highest number of potential anomalies with approximately 36,000 outliers, followed closely by SBERT_ARABERTv2_Custom with around 35,000. For the number of reduced outliers across different embedding models, SBERT_ARABERTv2 demonstrated the highest efficiency in detecting consolidated outliers (approximately 1500 instances), while FastText_Custom showed more conservative detection with around 400 reduced outliers. This substantial reduction in outliers suggests the effectiveness of data preprocessing techniques and the robustness of the embedding models. Moreover, the significant difference between raw and reduced outliers indicates the importance of our outlier consolidation process in refining the model's sensitivity.

These findings highlight the complementary strengths of different embedding approaches. While SBERT_ARABERTv2 excels in anomaly detection, FastText_Custom stands out for its superior topic coverage. FastText_Custom demonstrated balanced sensitivity to anomalies, detecting approximately 20,000 initial outliers and efficiently reducing them to 400 consolidated cases. This custom-trained model also offers better domain adaptation for traffic-related Arabic content compared to pre-trained alternatives, enabling more accurate capture of domain-specific nuances. Additionally, FastText's ability to handle out-of-vocabulary words through subword information proves particularly valuable for processing Arabic social media text, which often contains colloquial expressions and regional variations. Because of these reasons, we utilized FastText_Custom as the primary embedding model in our study.

Figure 4 presents a visual representation of the hierarchical clustering analysis across different topics. This analysis facilitated the detection of topic similarities and thematic relationships, enabling informed decisions about topic consolidation to align with our research goals. This systematic approach helped optimize the granularity of our topic classification by identifying and merging related themes, ensuring a balanced and meaningful topic structure.

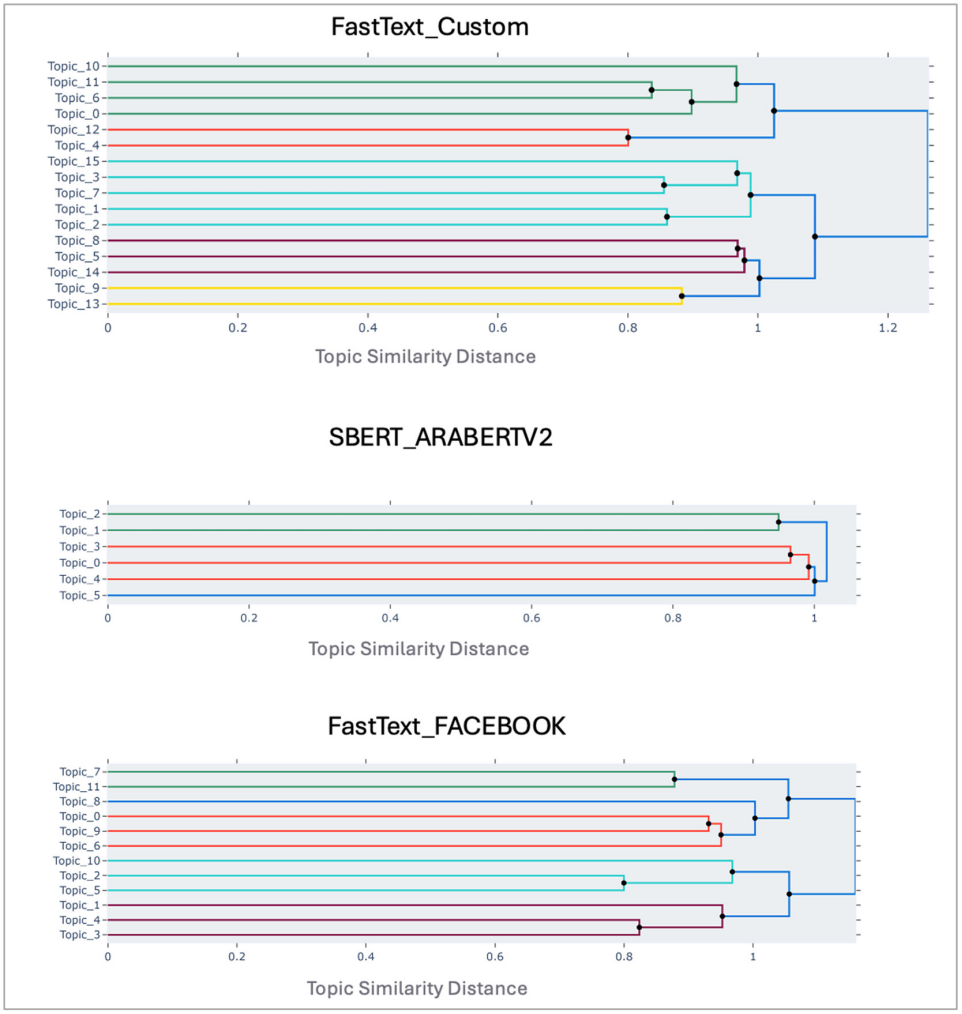


Figure 4. Hierarchical representation by using embedding models.

FastText_Custom demonstrated the most comprehensive topic clustering structure with 16 well-defined topics, showing clear hierarchical relationships and balanced cluster distances. The dendrogram suggests optimal topic separation, with clusters forming at

various distance levels, indicating nuanced relationships between related topics in traffic-related discussions. Next, SBERT_ARABERTv2 exhibited a more condensed clustering pattern with 6 topics, suggesting a higher-level categorization of content. While this might indicate less granular topic differentiation, it could be advantageous for broad-category sentiment analysis. The custom version of SBERT_ARABERTv2 showed improved topic granularity with 14 topics, indicating that customization enhanced its ability to detect subtle topic variations. FastText_Facebook identified 12 distinct topics, with clusters forming at similar distance levels, suggesting moderate topic differentiation capability. This analysis supports our choice of FastText_Custom as the primary embedding model, as it provided the most detailed and well-balanced topic clustering structure, essential for comprehensive sentiment analysis of traffic-related content.

4.2. Aspect Category Detection

The ACD applies topic modeling through BERTopic to cluster the topics within the data. The main component of BERTopic, called the 'approximate_distribution' algorithm, enables efficient handling of large datasets and robust topic distributions. The algorithm selects a random sample of documents from the dataset and calculates the topic distribution for this sample. It also identifies potential outlier documents with unusual topic distributions and adjusts their influence on the final model.

The word representation model used in this process is KeyBERTInspired, which identifies the most relevant words for each topic to capture the major themes within a document collection. BERTopic automatically determines the optimal number of topics using the HDBSCAN algorithm, which clusters the data into distinct groups (topics) in data. In our case, this resulted in 16 topics, each represented by the 10 most relevant words selected based on their importance to the topic. The top 10 words represented the key concerns and interests of social media users. Figure 5 shows categories and the leading 10 words associated with each category.

Notable topics include:

- License renewal (تجديد رخصة القيادة): A prominent topic with high word scores, indicating frequent discussions about driving license services and renewal procedures.
- Vehicle inspection services (الفحص الدوري للمركبة): Also showed significant representation, reflecting regular user engagement with vehicle testing and certification processes.
- Traffic violation inquiries (الاستعلام عن المخالفات المرورية): Demonstrated substantial user interest in checking and managing traffic penalties.
- Accident reporting (الابلاغ عن مشاكل الطرق): Showed active citizen participation in road safety monitoring.
- Vehicle registration (تسجيل المركبات) and plate inquiries (اسقاط المركبات): Reflecting user interest in vehicle documentation procedures.
- Appointment booking for driving schools (حجز موعد في مدرسة تعليم القيادة): A key service-related topic.

These findings demonstrate that the BERTopic successfully identified key traffic-related themes, providing valuable insights into user interactions with traffic services. The word scores indicate the relative importance of different aspects within each topic, helping prioritize service improvements. Additionally, Figure 6 visualizes tweet distribution across different topics. It reveals significant variations in user engagement across different traffic-related themes.

Road problems reporting emerged as the most discussed topic, with approximately 8000 posts, indicating its importance to citizens. Violation payment clearance and

Driving license issuance followed as the second and third most frequent topics, with around 5800 and 5400 mentions, respectively. Mid-range topics included vehicle deregistration, driving license renewal, and traffic violation objections, each garnering between 4100 and 4600 posts. Less frequently discussed topics included vehicle plate services, driving school booking, and fines reduction, with approximately 3400–3500 posts. This distribution provides valuable insights into user priorities and highlights areas of potential improvement.

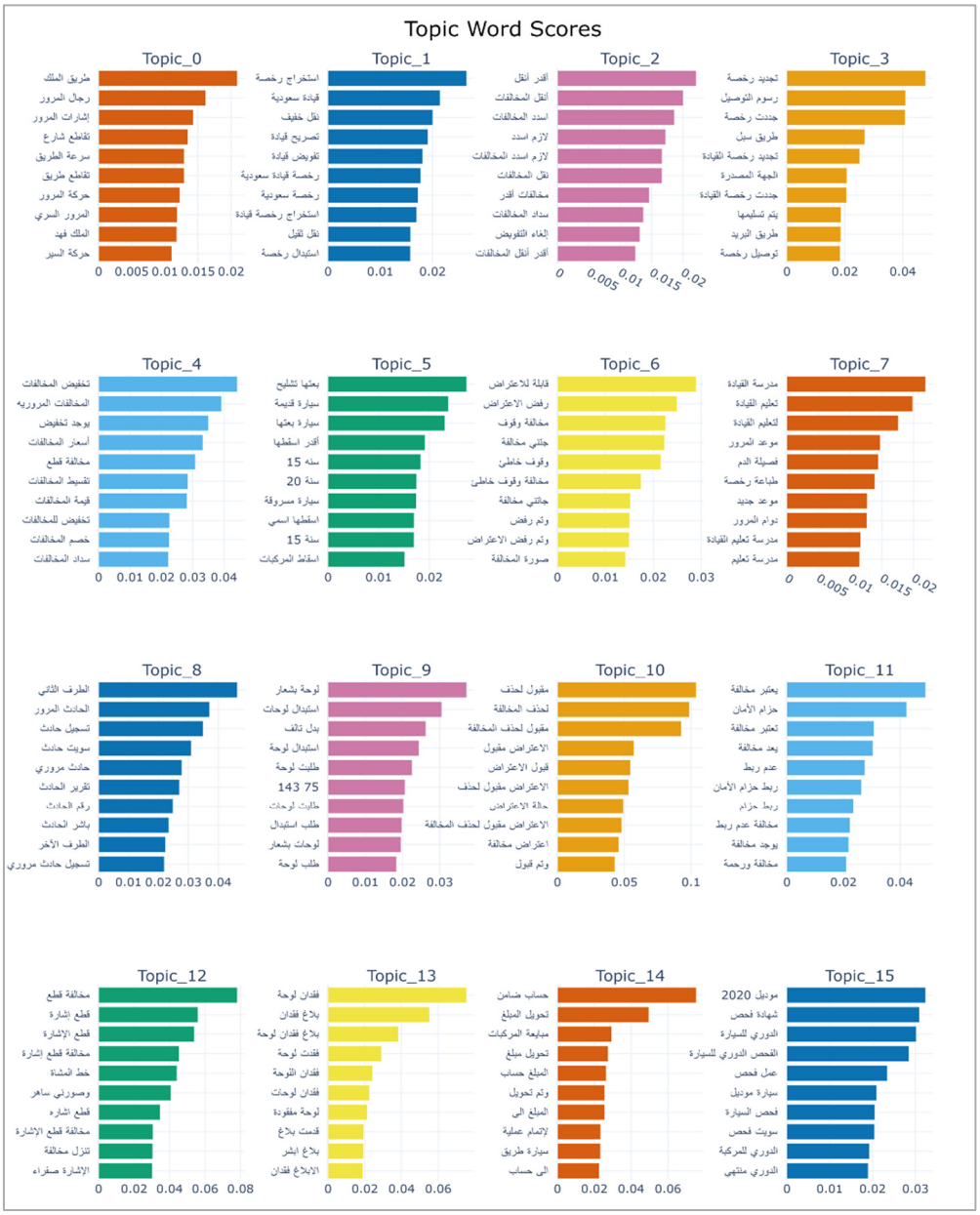


Figure 5. Topic words representation by BERTopic and FastText embedding.

We developed the CAMeLBERT model and trained it on the output of the BERTopic topic modeling. This approach eliminated the need for extensive manual annotation of training data. To optimize performance, we fine-tuned key hyperparameters, including learning rate, weight decay, warm-up ratio, and batch size. The objective was to reduce the loss function and enhance the framework's effectiveness on a separate validation collection.

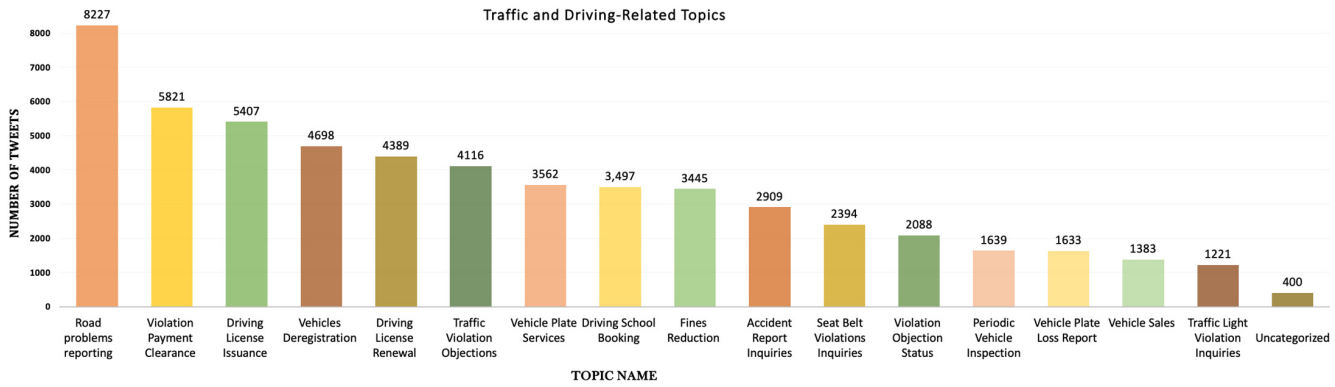


Figure 6. Number of tweets for each topic.

During training, we assessed the framework's effectiveness by tracking validation loss, which ranges from 0 to 1 and is calculated using cross-entropy. The validation loss, reported after each training epoch, helps evaluate the framework's ability to apply to new, unexplored data. A decrease in validation loss indicates fewer prediction errors and improved accuracy. We performed hyperparameter tuning to maximize effectiveness on an independent validation collection, using weighted cross-entropy loss with weights normalized to the inverse class distribution. The model demonstrated effective learning with validation loss stabilizing around 0.25, while training loss decreased significantly from 0.806 to 0.007. While the training experiment required 6 to 10 epochs, the model achieved optimal performance by epoch 6, with a 0.93 training accuracy score and the lowest validation loss. Next, we evaluated the framework's effectiveness on our designated test set and reported the model's performance as shown in Table 5 and Figure 7.

Table 5. Performance of CAMeLBERT_TOPICS_CLSSIFICATION_FOR_TRAFFIC_SERVICES model.

Topics	Precision	Recall	F1-Score
Driving License Issuance	0.9244	0.9464	0.9353
Road Problem Reporting	0.9581	0.9797	0.9688
Violation payment clearance	0.8923	0.8869	0.8896
Driving License Renewal	0.9342	0.9530	0.9435
Fines Reduction	0.9668	0.9129	0.9391
Submit A Request to Issue or Replace Vehicle Plates	0.9187	0.9559	0.9369
Driving School Booking	0.9038	0.8244	0.8623
Vehicles Deregistration	0.9431	0.9343	0.9387
Traffic Violation Objections	0.9128	0.9290	0.9208
Seat Belt Violations Inquiries	0.9461	0.9461	0.9461
An Accident Report Inquiries	0.8947	0.8889	0.8918
Violation Objection Status	0.9200	0.9293	0.9246
Periodic Vehicle Inspection	0.9022	0.9432	0.9222
Vehicle Sales	0.9848	0.9701	0.9774
Vehicle Plate Loss Report	0.9259	0.8621	0.8929
Traffic Light Violation Inquiries	0.8833	0.9138	0.8983
Accuracy			0.9282
Macro Avg	0.9257	0.9235	0.9243
Weighted Avg	0.9282	0.9282	0.9279

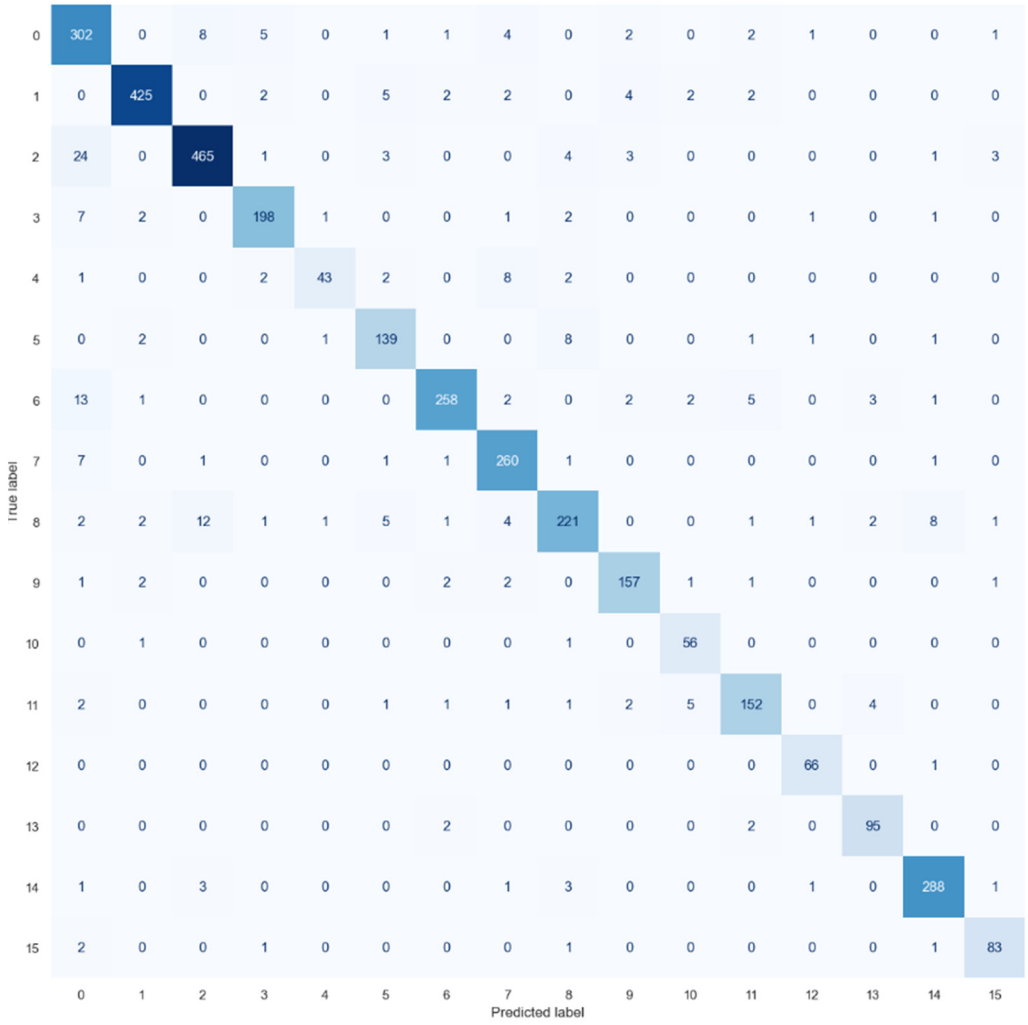


Figure 7. Confusion matrix of CAMeLBERT_TOPICS_CLSSIFICATION_FOR_TRAFFIC_SERVICES model.

Table 5 and Figure 7 present the prediction results of the CAMeLBERT model using a confusion matrix and performance metrics. The framework attained an average accuracy of 93% and a macro-averaged F1-score of 0.92. The results demonstrated the strong categorization capabilities of the model in 16 distinct traffic-related areas. Several categories achieved strong performance, such as "Vehicle Sales" and "Road problems reporting" receiving F1-scores of 0.98 and 0.97, respectively. With a weighted average F1-score of 0.93, the model performed consistently in the majority of categories. The model maintained strong performance with F1-scores above 0.89, even for categories with fewer samples, such as 'Vehicle Plate Loss Report' and 'Traffic Light Violation Inquiries' (approximately 58 samples). The model's resilience in managing class imbalance is demonstrated by the macro-average precision of 0.93 and recall of 0.92, showing balanced performance across all categories, irrespective of their size. The model performs consistently well across most service categories, as indicated by strong diagonal values in the confusion matrix, reflecting accurate predictions. The category "Driving License Issuance" had the highest number of samples (504), indicating a frequently requested service. The model's predictions for this category are more reliable due to the larger training data. This insight can help government agencies focus improvements on high-demand services.

Our CAMeLBERT-based topic classification model achieved remarkable performance, with an accuracy of 92.82% and a weighted F1-score of 92.79, significantly outperforming previous approaches in Arabic ACD. It is important to note that no previous studies have specifically addressed ACD within the Arabic traffic services domain, making our work

the first of its kind in this field. Furthermore, since our dataset is newly collected and specifically focused on traffic services, comparisons with previous works with different datasets are not possible. However, we can compare our model's performance to related works in other Arabic domains. Bensoltane and Zaki [3] achieved an F1-score of 65.5% using BiGRU with AraBERT embeddings on hotel reviews, while Almasri et al. (2023) [25] reported a micro F1-score of 0.682 using a semi-supervised approach with AraBERT on the benchmark datasets. In more recent studies, Ameur et al. (2023) [26] achieved an F1-score of 67.3% using AraBERT with multi-label learning for hotel reviews, and Fadel et al. (2024) reported an F1-score of 68.21% for ACD using their MTL-AraBERT model [27].

4.3. Aspect Polarity Classification

We fine-tuned and trained the CAMeLBERT model for our task of sentiment classification on the training collection and assessed it on the validation set. We observed that our data are highly imbalanced, comprising 75% natural, 23% negative, and 2% positive. To address this imbalance, we used a weighted cross-entropy loss function, assigning normalized inverse weights proportional to the class distribution. This approach incentivized the model to focus on minority classes during training.

Throughout five training epochs, the CAMeLBERT model showed remarkable performance in sentiment classification, exhibiting high accuracy and quick convergence. With an accuracy of 0.976 in the first epoch, the model demonstrated remarkable performance. It then improved further, reaching an accuracy of 0.995 by the fifth epoch. The validation loss indicated strong generalization, beginning at 0.120 and stabilizing around 0.128. Concurrently, the training loss decreased significantly from 0.327 to 0.003, indicating the model's consistent optimization.

Optimal performance was observed by the third epoch when the model achieved 0.99 training accuracy with the lowest validation loss. The optimal hyperparameter configurations were learning_rate = 1×10^{-5}, weight_decay = 1×10^{-3}, batch size = 32 for training and 64 for evaluation, and warmup_ratio = 0.05. Finally, we tested the model performance on the gold test and summarized the results in Table 6.

Table 6. Performance of CAMeLBERT_SENTIMENT_CLSSIFICATION_FOR_TRAFFIC_SERVICES model.

	Precision	**Recall**	**F1-Score**
Neutral	0.9962	0.9954	0.9958
Negative	0.9814	0.9851	0.9832
Positive	0.9149	0.8958	0.9053
Accuracy			0.9916
Macro Avg.	0.9642	0.9588	0.9614
Weighted Avg.	0.9916	0.9916	

The framework attained a weighted average F1-score of 0.991, indicating strong performance across all three sentiment categories. With an F1-score of 0.995 for neutral categories (2615 samples) and an F1-score of 0.983 for negative categories (803 samples), the model performed notably well. Despite the significant class imbalance, the model also performed commendably for positive opinions, achieving an F1-score of 0.905 from just 48 instances. The model's robustness in handling the unbalanced dataset while maintaining high classification accuracy is further illustrated by the high macro-average precision (0.964) and recall (0.958), which show balanced performance across all classes.

4.4. ABSA System Evaluation

We evaluated the performance of our integrated ABSA pipeline that combines two CamelBERT-based classifiers: ACD and sentiment polarity classification. The unified pipeline processes each input text through both classifiers independently, where one classifier identifies the service category (aspect) and the other determines the sentiment polarity. The system assigns each text a topic (e.g., Road problems reporting) and a sentiment (e.g., 'negative'), creating paired predictions like 'Report_Road_Problems#negative'.

Table 7 shows the evaluation metrics for these paired predictions, highlighting the system's performance across different topic-sentiment combinations. For example, the Road problems reporting category achieved an F1-score of 0.958 for negative sentiments, an F1-score of 0.878 for neutral sentiments, and F1 = 0.909 for positive sentiments, showing consistent performance across all sentiment types. In another category "Driving License Issuance", the neutral sentiments achieved an F1-score of 0.938 with 496 samples, while negative sentiments recorded an F1-score of 0.500 with only 8 samples, showing performance variation due to class distribution imbalance. The combined system maintained balanced performance across all aspect-sentiment combinations, attaining a macro-average F1-score of 0.831, demonstrating the effectiveness of using specialized classifiers for each task.

Table 7. Performance of the ABSA model.

ABSA System (Category#Sentiment)	Precision	Recall	F1-Score	Support
Driving_School_Booking#negative	0.8077	0.6176	0.7000	34
Driving_School_Booking#neutral	0.9151	0.8546	0.8838	227
Driving_School_Booking#positive	0.0000	0.0000	0.0000	1
Driving_License_Renewal#negative	0.8000	0.8696	0.8333	23
Driving_License_Renewal#neutral	0.9424	0.9562	0.9493	274
Driving_License_Renewal#positive	1.0000	1.0000	1.0000	1
Vehicles_Deregistration#negative	0.9222	0.9121	0.9171	91
Vehicles_Deregistration#neutral	0.9328	0.9250	0.9289	120
Vehicles_Deregistration#positive	1.0000	1.0000	1.0000	2
An_Accident_Report_Inquiries#negative	0.8551	0.8806	0.8676	67
An_Accident_Report_Inquiries#neutral	0.9036	0.8721	0.8876	86
Violation_Objection_Status#negative	0.9524	0.9091	0.9302	22
Violation_Objection_Status #neutral	0.9114	0.9351	0.9231	77
Fines_Reduction#negative	0.9500	0.7917	0.8636	24
Fines_Reduction#neutral	0.9680	0.9237	0.9453	262
Fines_Reduction#positive	1.0000	1.0000	1.0000	1
Seat_Belt_Violations_Inquiries#negative	0.7143	0.6250	0.6667	8
Seat_Belt_Violations_Inquiries#neutral	0.9434	0.9494	0.9464	158
Seat_Belt_Violations_Inquiries#positive	1.0000	1.0000	1.0000	1
Traffic_Light_Violation_Inquiries#negative	0.6667	0.8000	0.7273	10
Traffic_Light_Violation_Inquiries#neutral	0.9167	0.9167	0.9167	48
Driving_License_Issuance#negative	0.7500	0.3750	0.5000	8
Driving_License_Issuance#neutral	0.9238	0.9536	0.9385	496
Traffic_Violation_Objections#negative	0.9390	0.8851	0.9112	87
Traffic_Violation_Objections#neutral	0.8652	0.9390	0.9006	82

Table 7. *Cont.*

ABSA System (Category#Sentiment)	Precision	Recall	F1-Score	Support
Traffic_Violation_Objections#positive	0.0000	0.0000	0.0000	0
Periodic_Vehicle_Inspection#negative	0.3333	0.5000	0.4000	2
Periodic_Vehicle_Inspection#neutral	0.9213	0.9535	0.9371	86
Vehicle_Plate_Loss_Report#neutral	0.9535	0.8723	0.9111	47
Vehicle_Plate_Loss_Report#negative	0.8182	0.8182	0.8182	11
Road_Problems_Reporting#negative	0.9365	0.9806	0.9581	361
Road_Problems_Reporting#neutral	0.9231	0.8372	0.8780	43
Road_Problems_Reporting#positive	0.9459	0.8750	0.9091	40
Violation_Payment_Clearance#negative	0.8710	0.9000	0.8852	30
Violation_Payment_Clearance#neutral	0.8973	0.8851	0.8912	296
Violation_Payment_Clearance#positive	0.5000	1.0000	0.6667	1
Submit_A_Request_to_Issue_or_Replace _Vehicle_Plates#negative	0.8500	0.8947	0.8718	19
Submit_A_Request_to_Issue_or_Replace _Vehicle_Plates#neutral	0.9160	0.9524	0.9339	252
Submit_A_Request_to_Issue_or_Replace _Vehicle_Plates#positive	1.0000	1.0000	1.0000	1
Vehicle_Sales#negative	0.8571	1.0000	0.9231	6
Vehicle_Sales#neutral	1.0000	0.9672	0.9833	61
Accuracy			0.9207	3466
Macro Avg	0.8367	0.8373	0.8318	3466
Weighted Avg	0.9213	0.9207	0.9202	3466

The end-to-end ABSA system showed a remarkable accuracy of *92%* and demonstrated strong performance in many service categories. For the Vehicle Sales category, for example, it accomplished an F1-score of 0.983 for the neutral category and 0.923 for the negative category. For the 'Road problems reporting' category, it achieved F1-scores of 0.958, 0.878, and 0.909 for negative, neutral, and positive sentiments, respectively. Additionally, the system achieved a weighted average precision of 0.921 and recall of 0.920 across all categories and sentiment combinations, indicating balanced performance.

The system exhibited strong performance across different aspect-sentiment combinations, with F1-scores exceeding 0.85 in the majority. However, uncommon combinations, particularly those with few examples in the dataset, tended to perform worse. For instance, categories associated with positive sentiments often had fewer instances, which adversely affected their performance metrics. Despite these challenges, the system maintained strong macro-average metrics (precision = 0.836, recall = 0.837, F1-score = 0.831), suggesting its resilience throughout the whole spectrum of traffic-related service categories and sentiment combinations. Additionally, the system's ability to handle the natural class distribution inherent in real-world traffic service applications is confirmed by the high weighted averages across all measures (approximately 0.922).

The combination 'Traffic_Violation_Objections#positive', which achieved 0 for precision, recall, and F1-score in the test set, is a noteworthy finding. Although this combination was present in the training and validation collections, it was absent in the test set. This absence does not imply that such examples do not exist in the domain but rather highlights a flaw in our assessment design.

To address this issue, future work must prioritize ensuring a balanced distribution of data for all splits (training, validation, and test), particularly for rare aspect-sentiment combinations. Imbalanced splits can affect the ability to accurately assess the model's performance on all potential combinations, even when the model is trained to handle them.

For extremely rare aspect-sentiment combinations (those with only 1–2 examples in the test set), the evaluation revealed significant patterns. Some uncommon combinations, such as "Driving_License_Renewal#positive" and "Vehicles_Deregistration #positive", achieved perfect performance (F1 = 1.000). However, others, like "Driving_School_Booking#positive" (F1 = 0.000) and "Periodic_Vehicle_Inspection#negative" (F1 = 0.400), exhibited poor performance. This substantial variation in classification performance for cases with low sample sizes underscores the limitations of evaluation metrics when applied to extremely small datasets. These findings suggest that future iterations of this work would benefit from targeted data collection efforts to ensure a more balanced representation across all aspect-sentiment combinations. Such efforts would enhance the reliability and robustness of performance assessments, particularly for underrepresented categories.

4.5. ATS Models

We conducted extensive experiments using two fine-tuned models, AraBART and mT5, to generate summaries explaining the reasons behind positive or negative sentiments about traffic services for each topic. Both models were fine-tuned on our specialized dataset of user tweets related to traffic services. For an objective evaluation of their performance, we followed standard practices in the field and utilized ROUGE metrics as our primary evaluation framework [28]. This approach allowed us to assess the model's ability to comprehend and concisely summarize traffic-related content in Arabic social media posts.

Table 8 shows the outcomes of the training process of the AraBART and mT5 models over five epochs, revealing significant improvements in their performance. Both models achieved high scores of 0.80 on the ROUGE-1, ROUGEL, and ROUGESUM evaluation metrics, indicating they perform equally well on these summarization benchmarks. While mT5 is a multilingual model, AraBART is fine-tuned for the Arabic language. This fine-tuning likely contributes to its strong performance on Arabic text summarization tasks, matching the results of the more general-purpose mT5 model. For the ROUGE-L metric, both AraBART and mT5 obtained a score of 0.80, again showing similar performance. Overall, the table suggests that the AraBART and mT5 models exhibited comparable results across the range of ROUGE evaluation metrics presented. This implies that both models demonstrated strong ATS capabilities.

Table 8. Training results of Arabic abstractive summarization models.

Model Training	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
AraBART	0.80	0.76	0.80	0.80
mT5	0.80	0.76	0.80	0.80

During the training of the AraBART model, the training loss started at 1.0582 in epoch 2 and decreased significantly to 0.3681 by epoch 5, indicating the framework's successful learning during the training process. Similarly, the validation loss follows a downward trend, starting at 0.534543 in epoch 2 and decreasing slightly to 0.530071 by epoch 5. This indicates that the framework generalized well to unexplored validation data. All ROUGE metrics scores showed consistent improvement across the 5 epochs, with the final epoch achieving scores of 0.80 (ROUGE-1), 0.76 (ROUGE-2), 0.80 (ROUGE-L), and 0.80 (ROUGE-

Lsum). These high ROUGE scores indicate the framework's ability to produce high-quality abstractive text summaries, confirming its suitability for Arabic text summarization tasks. The results suggest that AraBART developed effective summarization capabilities during the training phase.

Despite the notable performance trends observed during training, both fine-tuning AraBART and mT5 models achieved remarkably similar performance levels on the test collection as shown in Table 9. AraBART achieved scores of 0.80 for ROUGE-1, 0.75 for ROUGE-2, and 0.79 for ROUGE-L and ROUGE-Lsum, while mT5 showed nearly identical performance with 0.79 for ROUGE-1, 0.75 for ROUGE-2, and 0.78 for ROUGE-L and ROUGE-Lsum, suggesting that both models developed comparable generalization capabilities in understanding and summarizing Arabic traffic-related feedback from social media.

Table 9. Evaluation results of Arabic summarization models.

MODEL	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
AraBART	0.80	0.75	0.79	0.79
mT5	0.79	0.75	0.78	0.78

The high ROUGE-2 scores of 0.75 for both models indicate exceptional ability to maintain proper phrasing and local coherence when summarizing user's sentiment. Additionally, both models successfully addressed Arabic-specific challenges such as dialectal variations, traffic terminology, and informal social media language while preserving the core meaning of user complaints, suggestions, and observations about traffic services.

Despite the challenging nature of Arabic social media text, which often includes dialectal variations and domain-specific terminology, both AraBART and mT5 demonstrated strong summarization capabilities. The nearly identical performance of both models suggests that either can be effectively deployed for summarizing Arabic traffic-related user sentiments, with the choice dependent on practical considerations such as computational resources or deployment constraints.

4.6. Comparison with State-of-the-Art Models

Our CAMeLBERT models for Arabic text classification in the traffic services domain achieved remarkable results compared to previous benchmarks and studies. For Aspect polarity classification, our model attained 99.16% accuracy and 96.14% macro-average F1-score, with class-specific F1-scores of 99.58% for neutral, 98.32% for negative, and 90.53% for positive sentiments. This represents a substantial improvement over the previous work by Almutrash & Abudalfa [46], which achieved only 63.87% accuracy and 63.46% macro F1-score using the AT-ODTSA dataset. It also significantly outperforms the original CAMeLBERT benchmarks reported by Inoue et al. [45], which showed F1 scores of 76.9% on ASTD, 93.0% on ArSAS, and 72.1% on SemEval datasets. Similarly, our aspect category detection model for traffic services demonstrated strong performance with 92.82% accuracy and 92.43% macro-average F1 score across 16 different service categories. Individual topic F1 scores ranged from 86.23% (Driving School Booking) to 97.74% (Vehicle Sales), with most topics exceeding 90%. Little work has been performed for aspect category detection to compare it with. Only one work includes the original CAMeLBERT's performance on poetry classification (APCD dataset) [45], which achieved a maximum F1 score of 80.9%. Our results demonstrate that CAMeLBERT models have shown promising results across various Arabic NLP tasks.

For abstractive text summarization, we benchmarked our approach against recent Arabic text summarization models. Our fine-tuned AraBART model achieved scores of

0.80, 0.75, and 0.79 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, significantly outperforming previous implementations, including:

- Compared to ROUGE-1: 0.424, ROUGE-2: 0.288, and ROUGE-L: 0.403 by Eddine et al. [28], our model showed improvements of approximately 37.6, 46.2, and 38.7 percentage points, respectively.
- Compared to ROUGE-1: 0.55, ROUGE-2: 0.402, and ROUGE-L: 0.546 by Alqahtani et al. [29], our model demonstrated gains of 25, 34.85, and 24.45 percentage points, respectively.
- Compared to ROUGE-1: 0.25, ROUGE-2: 0.12, and ROUGE-L: 0.25 by Masri et al. [30], remarkable improvements of 55, 63, and 54 percentage points, respectively, were observed.

Similarly, our fine-tuned mT5 model achieved impressive scores of 0.79, 0.75, and 0.78 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, outperforming previous mT5-based implementations including:

- Compared to ROUGE-1: 0.76, ROUGE-2: 0.62, and ROUGE-L: 0.75 by Elsaid et al. [4]; our model demonstrated improvements of 3, 13, and 3 percentage points, respectively.
- Compared to ROUGE-1: 0.06, ROUGE-2: 0.00, and ROUGE-L: 0.06 by Masri et al. [30], our approach achieved remarkable gains of 73, 75, and 72 percentage points, respectively.

These comprehensive improvements across all benchmarks and models suggest that our approach, combining optimized model architecture, effective preprocessing and hyperparameter tuning, and refined training methodology, represents a substantial advancement in Arabic abstractive summarization capabilities.

These results establish leading-edge performance standards for Arabic Aspect-based sentiment analysis and abstractive summarization in the traffic domain, demonstrating the potential of transformer-based architectures for processing and analyzing Arabic social media content in specialized domains.

5. Conclusions and Future Work

This research introduced a novel unsupervised approach for ABSA and ATS in the traffic services domain. We observed several notable results. First, this study provided compelling evidence that domain-specific embedding models significantly outperform general-purpose Arabic language models. Specifically, FastText custom embeddings achieved superior topic clustering with 16 distinct service categories, outperforming pre-trained alternatives. Second, the results demonstrated the effectiveness of unsupervised approaches in handling the complexity of Arabic social media content without requiring extensive manual annotation, challenging the conventional reliance on supervised learning in ABSA and ATS tasks. Third, the integrated ABSA system with ACD and APC exhibited strong performance across various aspect-sentiment combinations. It achieved a macro-average F1-score of 0.831, demonstrating robust handling of Arabic dialectal variations and domain-specific terminology. Additionally, the study also revealed that transformer-based models can successfully adapt to Arabic dialectal variations and domain-specific terminology through careful fine-tuning. Notably, AraBART demonstrated superior performance in summarization tasks, achieving ROUGE scores of 0.80, 0.75, and 0.79 for ROUGE-1, ROUGE-2, and ROUGE-L, respectively, establishing new benchmarks for summarizing Arabic traffic services.

Despite these achievements, our research encountered several significant challenges and limitations. A key issue was data representation within the test set, where certain aspect-sentiment combinations were absent, such as 'Traffic_Violation_Objections#positive',

leading to zero performance metrics. This gap hindered comprehensive performance evaluation for all combinations. Additionally, a high volume of outliers (approximately 36,000) in topic modeling required specialized reduction techniques and sophisticated consolidation processes to achieve meaningful topic classifications.

Language variation posed a significant challenge, as the system needed to simultaneously handle multiple Arabic dialects, informal language patterns, and domain-specific terminology common in processing social media content. This complexity was further compounded by the diverse ways users express their opinions about traffic services. Furthermore, maintaining consistent system performance across different types of user opinions proved challenging.

While the system performed well on common categories and sentiment combinations and achieved high F1 scores for well-represented classes, performance varied significantly for rare combinations. Some categories achieved perfect F1 scores (1.0), whereas others performed poorly (F1 = 0), highlighting the limitations of current methods in handling underrepresented categories.

These findings emphasize the need for more balanced data collection strategies and robust approaches for handling underrepresented categories. Future investigations could examine various directions to overcome these limitations. First, developing more sophisticated methods to handle rare aspect-sentiment combinations is crucial. Second, investigating multi-task learning architectures could improve efficiency and performance across all three tasks—ACD, APC, and ATS—simultaneously. Third, analyzing emerging topics in real time enables traffic service agencies to proactively address user concerns and improve service delivery.

Author Contributions: Conceptualization, A.A. and F.N.; supervision, F.N.; data collection and analysis, A.A.; methodology, A.A. and F.N.; formal analysis, A.A.; original draft writing, A.A.; review and editing, F.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from author Alanoud Alotaibi, upon reasonable request.

Acknowledgments: The authors gratefully thank the General Department of Traffic (GDT) and King Abdulaziz University (KAU). In addition, we would like to express our sincere gratitude to all participants who generously contributed their time and effort in the data labeling process. Their careful attention to detail and dedication were essential in creating a high-quality dataset for this research. The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU) for funding and supporting this work through the Graduate Students Research Support Program (IMSIU-GSRSP).

Conflicts of Interest: The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Abdelgwad, M.M.; Soliman, T.H.A.; Taloba, A.I. Arabic aspect sentiment polarity classification using BERT. *J. Big Data* **2022**, *9*, 115. [CrossRef]
2. Ullah Khan, M.; AbdulRehman, J.; Ihsan, M.; Tariq, U. A Novel Category Detection of Social Media Reviews in the Restaurant Industry. *Multimed. Syst.* **2023**, *29*, 1825–1838. [CrossRef]
3. Bensoltane, R.; Zaki, T. Comparing word embedding models for Arabic aspect category detection using a deep learning-based approach. *E3S Web Conf.* **2021**, *297*, 01072. [CrossRef]
4. Elsaid, A.; Mohammed, A.; Fattouh, L.; Sakre, M. Abstractive Arabic Text Summarization Based on MT5 and AraBart Transformers. In Proceedings of the 1st International Conference of Intelligent Methods, Systems and Applications (IMSA 2023), Giza, Egypt, 15–16 July 2023; pp. 7–12. [CrossRef]

5. Bensoltane, R.; Zaki, T. Towards Arabic aspect-based sentiment analysis: A transfer learning-based approach. *Soc. Netw. Anal. Min.* **2022**, *12*, 7. [CrossRef]
6. Al-Twairesh, N.; Al-Khalifa, H.; Al-Salman, A. Subjectivity and Sentiment Analysis of Arabic: Trends and Challenges. In Proceedings of the IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA), Doha, Qatar, 10–13 November 2014; pp. 148–155. [CrossRef]
7. Al-Smadi, M.; Qawasmeh, O.; Talafha, B.; Quwaider, M. Human Annotated Arabic Dataset of Book Reviews for Aspect Based Sentiment Analysis. In Proceedings of the International Conference on Future Internet of Things and Cloud (FiCloud), Rome, Italy, 10–12 October 2015; pp. 726–730. [CrossRef]
8. Pontiki, M.; Papageorgiou, H.; Galanis, D.; Androutsopoulos, I.; Pavlopoulos, J.; Manandhar, S. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland, 23–24 August 2014; pp. 27–35.
9. Al-Ayyoub, M.; Al-Sarhan, H.; Al-So'ud, M.; Al-Smadi, M.; Jararweh, Y. Framework for affective news analysis of Arabic news: 2014 Gaza attacks case study. In Proceedings of the 7th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 5–7 April 2016; pp. 327–332.
10. Areed, S.; Alqaryouti, O.; Siyam, B.; Shaalan, K. Aspect-Based Sentiment Analysis for Arabic Government Reviews. In *Studies in Computational Intelligence*; Springer: Cham, Switzerland, 2020; Volume 874, pp. 143–162. [CrossRef]
11. Al-Smadi, M.; Talafha, B.; Al-Ayyoub, M.; Jararweh, Y. Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 2163–2175. [CrossRef]
12. Al-Smadi, M.; Qwasmeh, O.; Talafha, B.; Al-Ayyoub, M.; Jararweh, Y.; Benkhelifa, E. An Enhanced Framework for Aspect-Based Sentiment Analysis of Hotels' Reviews: Arabic Reviews Case Study. Available online: http://alt.qcri.org/semeval2016/task5/ (accessed on 21 February 2025).
13. Pontiki, M.; Galanis, D.; Papageorgiou, H.; Androutsopoulos, I.; Manandhar, S.; Al-Smadi, M.; Al-Ayyoub, M.; Zhao, Y.; Qin, B.; De Clercq, O.; et al. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, CA, USA, 16–17 June 2016; pp. 19–30. [CrossRef]
14. Kumar, A.; Saini, M.; Sharan, A. Aspect category detection using statistical and semantic association. *Comput. Intell.* **2020**, *36*, 1161–1182. [CrossRef]
15. Ashi, M.; Siddiqui, M.; Nadeem, F. Pre-trained Word Embeddings for Arabic Aspect-Based Sentiment Analysis of Airline Tweets. In Proceedings of the International Conference on Advanced Intelligent Systems and Informatics, Cairo, Egypt, 1–3 September 2018; pp. 241–251.
16. Alshaikh, K.A.; Almatrafi, O.A.; Abushark, Y.B. BERT-Based Model for Aspect-Based Sentiment Analysis for Analyzing Arabic Open-Ended Survey Responses: A Case Study. *IEEE Access* **2024**, *12*, 2288–2302. [CrossRef]
17. Alshammari, N.; AlMansour, A. Aspect-based Sentiment Analysis for Arabic Content in Social Media. In Proceedings of the International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Istanbul, Turkey, 12–13 June 2020; pp. 1–6. [CrossRef]
18. Alyami, S.; Alhothali, A.; Jamal, A. Systematic literature review of Arabic aspect-based sentiment analysis. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 6524–6551. [CrossRef]
19. Pratt, L. Discriminability-Based Transfer between Neural Networks. *Adv. Neural Inf. Process. Syst.* **1992**, *5*, 204–211.
20. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models are Unsupervised Multitask Learners. *OpenAI Blog* **2019**, *1*, 9.
21. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.; Le, Q.V. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 5753–5763.
22. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
23. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised Cross-lingual Representation Learning at Scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 8440–8451.
24. Chouikhi, H.; Alsuhaibani, M.; Jarray, F. BERT-Based Joint Model for Aspect Term Extraction and Aspect Polarity Detection in Arabic Text. *Electronics* **2023**, *12*, 515. [CrossRef]
25. Almasri, M.; Al-Malki, N.; Alotaibi, R. A semi-supervised approach to Arabic aspect category detection using Bert and teacher-student model. *PeerJ Comput. Sci.* **2023**, *9*, e1425. [CrossRef] [PubMed]
26. Ameur, A.; Hamdi, S.; Ben Yahia, S. Multi-Label Learning for Aspect Category Detection of Arabic Hotel Reviews Using AraBERT. In Proceedings of the International Conference on Agents and Artificial Intelligence (ICAART), Lisbon, Portugal, 22–24 February 2023; pp. 241–250. [CrossRef]

27. Fadel, A.; Saleh, M.; Salama, R.; Abulnaja, O. MTL-AraBERT: An Enhanced Multi-Task Learning Model for Arabic Aspect-Based Sentiment Analysis. *Computers* **2024**, *13*, 98. [CrossRef]
28. Kamal Eddine, M.; Tomeh, N.; Habash, N.; Le Roux, J.; Vazirgiannis, M. AraBART: A Pretrained Arabic Sequence-to-Sequence Model for Abstractive Summarization. In Proceedings of the Seventh Arabic Natural Language Processing Workshop (WANLP), Abu Dhabi, United Arab Emirates, 8 December 2022; pp. 31–42.
29. Alqahtani, D.; Al-Yahya, M. Exploring Arabic Pre-Trained Language Models for Arabic Abstractive Text Summarization. In Proceedings of the 10th International Conference on Social Networks Analysis, Management and Security (SNAMS 2023), Online, 21–24 November 2023. [CrossRef]
30. Masri, S.; Raddad, Y.; Khandaqji, F.; Ashqar, H.I.; Elhe-Nawy, M. Transformer Models in Education: Summarizing Science Textbooks with AraBART, MT5, AraT5, and mBART. *Preprints*, 2023; *in press*.
31. Einieh, Y.; Almansour, A.; Jamal, A. Fine Tuning an AraT5 Transformer for Arabic Abstractive Summarization. In Proceedings of the 14th IEEE International Conference on Computational Intelligence and Communication Networks, Online, 4–6 December 2022. [CrossRef]
32. Elhassan, N.; Al-Sammarraie, N.; Kamel, N.; Al-Zaben, N.; Hamdeh, M.A.; Al Qatawneh, S.; Shatnawi, M.; Abu-Hassan, N. Arabic Sentiment Analysis Based on Word Embeddings and Deep Learning. *Computers* **2023**, *12*, 126. [CrossRef]
33. Facebook AI Research. word2vec Embedding Model. Available online: https://fasttext.cc/docs/en/crawl-vectors.html (accessed on 22 March 2025).
34. Amdalotaibi. Fastext_Custom Word Embedding Model. Available online: https://huggingface.co/Amdalotaibi/Fastext_Custom/tree/main (accessed on 22 March 2025).
35. Reimers, N.E.C.A. Sentence Embedding Distil-USE Model. Available online: https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1 (accessed on 22 March 2025).
36. MIND LAB. Sentence Embedding SBERT_ARABERTv2 Model. Available online: https://huggingface.co/aubmindlab/bert-base-arabertv2 (accessed on 22 March 2025).
37. Amdalotaibi. arabertv2-mlm-Finetuned-Traffics Model. Available online: https://huggingface.co/Amdalotaibi/arabertv2-mlm-finetuned-traffics-2 (accessed on 22 March 2025).
38. George, L.; Sumathy, P. An integrated clustering and BERT framework for improved topic modeling. *Int. J. Inf. Technol.* **2023**, *15*, 2187–2195. [CrossRef] [PubMed]
39. Pattanayak, P.K.; Tripathy, R.M.; Padhy, S. A semi-supervised approach of short text topic modeling using embedded fuzzy clustering for Twitter hashtag recommendation. *Discov. Sustain.* **2024**, *5*, 218. [CrossRef]
40. Abuzayed, A.; Al-Khalifa, H. BERT for Arabic Topic Modeling: An Experimental Study on BERTopic Technique. In *Procedia Computer Science*; Elsevier: Amsterdam, The Netherlands, 2021; Volume 189, pp. 191–194. [CrossRef]
41. Aouichaty, S.; Maleh, Y.; Mohtadi, M.T.; Hajami, A.; Allali, H. Sustainable topic modeling for legal Moroccan Arabic language: A challenging study on bertopic technique. In *Procedia Computer Science*; Elsevier B.V.: Amsterdam, The Netherlands, 2024; pp. 582–588. [CrossRef]
42. Alhaj, F.; Al-Haj, A.; Sharieh, A.; Jabri, R. Improving Arabic Cognitive Distortion Classification in Twitter Using BERTopic. Available online: www.ijacsa.thesai.org (accessed on 22 March 2025).
43. Uthirapathy, S.E.; Sandanam, D. Topic Modelling and Opinion Analysis on Climate Change Twitter Data Using LDA and BERT Model. In *Procedia Computer Science*; Elsevier B.V.: Amsterdam, The Netherlands, 2023; pp. 908–917. [CrossRef]
44. Nadim, M.; Akopian, D.; Matamoros, A. A Comparative Assessment of Unsupervised Keyword Extraction Tools. *IEEE Access* **2023**, *11*, 144778–144798. [CrossRef]
45. Inoue, G.; Alhafni, B.; Baimukan, N.; Bouamor, H.; Habash, N. The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models. *arXiv*, 2023; *preprint*.
46. Almutrash, S.; Abudalfa, S. Comparative Study on the Efficiency of Using PaLM and CAMeLBERT for Arabic Entity Sentiment Classification. In Proceedings of the SaudiCIS 2024 Proceedings, Dhahran, Saudi Arabia, 19–21 November 2024; pp. 11–19. Available online: https://aisel.aisnet.org/saudicis2024/66 (accessed on 22 March 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.