

A. Savostin¹,
orcid.org/0000-0002-5057-2942,
G. Kaipbek^{*2},
orcid.org/0000-0003-2595-7434,
K. Koshekov²,
orcid.org/0000-0002-9586-2310,
G. Savostina¹,
orcid.org/0000-0001-7042-4480,
K. Wardle^{2,3},
orcid.org/0009-0008-4866-3934

1 – Manash Kozybayev North Kazakhstan University, Petropavlovsk, Republic of Kazakhstan
2 – Civil Aviation Academy, Almaty, Republic of Kazakhstan
3 – JSC Air Astana, Almaty, Republic of Kazakhstan
* Corresponding author e-mail: kaipbekgulsanat@gmail.com

COMPREHENSIVE ANALYSIS OF AVIATION MAINTENANCE TEXT REPORTS USING NATURAL LANGUAGE PROCESSING METHODS

Purpose. This study aims to develop and validate a comprehensive approach for analyzing unstructured textual descriptions of defects extracted from actual aviation maintenance data. The goal is to improve both the efficiency and depth of fault analysis by addressing two key tasks: automatic classification of defects into standard categories and identification of latent thematic subgroups within these categories.

Methodology. The research is based on a dataset containing maintenance records from nine commercial aircraft over a seven-year period. A multi-stage preprocessing pipeline was developed, including an algorithm for domain-specific abbreviation identification and expert-driven decoding. To solve the multiclass classification task across 30 Chapter–Section (CS) categories, four approaches were compared: CountVectorizer with LinearSVC, TF-IDF and Word2Vec with logistic regression, and fine-tuning of the transformer-based DistilBERT model. For an in-depth analysis of the largest defect category, topic modeling based on Latent Dirichlet Allocation (LDA) was applied, with a quantitative procedure for selecting the optimal number of topics.

Findings. The best performance in classification was achieved by the TF-IDF with logistic regression approach, reaching $f1\text{-macro} = 0.762$ and Cohen's Kappa = 0.809, statistically comparable to CountVectorizer with LinearSVC. Classical methods significantly outperformed neural network models, underscoring their robustness for analyzing short technical texts. Topic modeling successfully decomposed the largest defect category into five interpretable and semantically coherent subgroups.

Originality. The novelty of this work lies in developing and testing a formalised method for analysing unstructured aviation maintenance data, implemented as a single integrated process. The study also provides a detailed comparative evaluation of classical and modern NLP models on domain-specific aviation maintenance data.

Practical value. The work is practical in nature and contains results which are ready for implementation. A prototype of an automated classifier has been created which is capable of processing the main flow of daily defect reports, reducing the time required for manual processing. An in-depth failure analysis tool has also been developed, which provides a transition from general fault codes to the analysis of specific sub-problems. This contributes to optimizing maintenance programs, enhancing diagnostic procedures, and ultimately improving flight safety.

Keywords: *natural language processing, maintenance, aircraft, classification, topic modeling*

Introduction. An aircraft, as an extremely complex, high-tech system, is susceptible to malfunctions caused by a variety of factors including human error, material defects, manufacturing errors, and extreme operating conditions [1]. Therefore, ensuring flight safety is fundamental to the development of civil aviation. Despite decades of work in this area, it is still impossible to eliminate existing risks completely. However, in recent years, significant progress has been made in improving the safety and economic efficiency of aircraft operations owing to the introduction of advances in computing, information and communication technologies, and machine learning [2, 3].

Many modern approaches in the aviation industry have focused on the implementation of diagnostic monitoring, failure prediction, and equipment condition assessment based on the intelligent analysis of structured data obtained from a large number of aircraft sensors and systems [4], including in real time [5].

Commercial aircraft maintenance also actively leverages these innovations by implementing predictive methods [6, 7]. However, despite this progress, most existing maintenance systems focus primarily on numerical data and missing valuable information contained in unstructured text reports.

Every day, aircraft maintenance and operation generate a significant volume of text data containing detailed descriptions of malfunctions, their symptoms, and repair actions. These records, created by experienced technical personnel and flight crews, represent a vast source of information that is often unnoticed by automated diagnostic systems oriented toward structured data [8]. Ignoring this information can lead to an incomplete understanding of the nature of failures in the ever-growing volume of data, thereby limiting the effectiveness of existing approaches to maintenance and reliability management.

It can be argued that the use of automatic analysis of text descriptions of defects will help avoid the subjectivity and potential biases inherent in manual categorization [9]. The intelligent analysis of maintenance records

can facilitate the early detection of safety issues and the extraction of hidden problems that may not be obvious using traditional approaches [10].

Although natural language processing (NLP) methods have demonstrated impressive results in various tasks, their direct application to aviation maintenance records presents unique challenges. This is because current advances in NLP primarily stem from the analysis of text datasets outside technical disciplines [11]. Although strategies such as domain adaptation and transfer learning offer the potential to extend the application of NLP to a variety of information streams, their success depends on the availability of topic-matched, well-resourced, and annotated data.

These conditions are not always applicable to the specialized domain of aviation maintenance because of their unique characteristics [8]. Text descriptions of defects and repairs are mostly unstructured and short, and contain a wealth of specialized vocabulary, abbreviations, codes, and professional jargon not found in general language models. This specificity requires a special approach that extends beyond standard NLP tools.

The use of large language models (LLM) for analyzing aircraft maintenance records also has significant limitations. Key issues include: insufficient interpretability of LLM output, which is critical for aviation with its requirement for traceability of decisions [12]; the tendency of LLM to “hallucinations” (generating unreliable information) [13]; and difficulties in ensuring the reproducibility and stability of results [14]. Additional challenges include privacy risks when using cloud-based application programming interfaces (APIs), the need for significant computational resources for effective LLMs, and the limited context window of existing models.

Thus, given the increasing complexity of aircraft technology and the volume of maintenance data generated, as well as the limitations of existing NLP approaches for aviation texts, there is an urgent need to develop practically applicable and interpretable analysis methodologies. There is a clear need for comprehensive solutions that not only automate the classification of defects into standard categories (e.g., aircraft systems) but also provide a deeper understanding of common patterns and specific subthemes of malfunctions within categories.

Such comprehensive analytical strategies open up the prospect of significantly improving maintenance processes and consequently increasing flight safety.

In addition to improving reliability and safety, it is important to consider the economic value of reducing the maintenance time. In aviation, maintenance delays lead to significant losses; therefore, speeding up diagnostics is critical [15]. Developing corrective actions requires a rapid response, and diagnostic errors can delay the repair. In such circumstances, technical language analysis methods help automate routine tasks and quickly provide necessary information.

Based on the above, this study proposes and tests a comprehensive analysis of textual reports on aviation defects using natural language processing methods to overcome existing difficulties in the industry.

The current state of the issue. The stated relevance of the task of analyzing text data in the aviation domain is reflected in a number of scientific publications and practical developments. Traditionally, researchers have fo-

cused on the automatic analysis of information contained in the Aviation Safety Reporting System (ASRS) [16]. For example, in [17], the Singular Value Decomposition (SVD) algorithm was used together with NLP to analyze incidents involving dangerous goods. In the study [18], structural topic modeling (STM) methods were applied to the reports of the ASRS and the National Transportation Safety Board (NTSB) [19], allowing for the identification of non-obvious patterns in topics. Deep-learning methods in NLP are also actively used in processing aviation safety reports. The authors of [20] used a Bidirectional Encoder Representations from Transformers (BERT) model to extract relevant information from tests. In addition, deep neural networks with a Long Short-Term Memory (LSTM) structure can be used to identify causal factors of aviation incidents, as shown in [21].

Overall, [22] provides a detailed analysis of existing publications on the application of machine learning (ML) and NLP to aviation safety text analysis. Consequently, the authors unequivocally recommend integrating NLP into the aviation industry’s safety management system. They also emphasize the importance of the interpretability of NLP analysis results and the need to develop guidelines for preparing and annotating text data.

Text pre-processing and annotation tasks were considered in [23]. The authors of this paper propose an effective two-stage training approach using Transformers and Sequential Denoising AutoEncoder (TSDAE), and Sentence BERT models for working with aviation-domain text datasets. The Digital Automatic Terminal Information Service (DATIS), which is weakly related to specific maintenance records, was selected as the primary source of training data in this study. Furthermore, the methods proposed in this study remain poorly interpreted.

The problem of preparing text data for NLP was thoroughly studied in [8]. The authors proposed a framework for processing technical language from the aviation domain for prediction and condition management (PHM) tasks. The study used the MaintNet dataset [24] with 6,169 records on aviation maintenance. It should be noted that this publication does not provide the quality metrics of the models synthesized by the authors for the task of predicting maintenance actions based on NLP, which significantly complicates the understanding of the significance of the achieved results. In addition, numerical metrics for the proposed method for the automatic classification of defects by ATA 100 codes (ATA iSpec 2200), described in the study [25] are not presented. The authors noted that the anonymized database used [24] could belong to a small or training aircraft, which is why the topology of the detected defects was significantly narrowed.

In [26], a model for classifying six types of aircraft failure based on text descriptions of defects was proposed. The classifier developed by the authors demonstrated high accuracy ($f1\text{-score} = 0.968$) on a dataset of 1,679 records of Chinese and English texts. The authors of [27] also limited themselves to 10 types of malfunctions and proposed a model for classifying defects based on aircraft maintenance records in Chinese.

In [28], researchers used a comprehensive dataset compiled from various sources to obtain risk ratings and

identify similar predictive indicators for aircraft defects using NLP. In this study, the authors used ATA 100 (ATA iSpec 2200) code to demonstrate the advantages of the proposed NLP methods.

Thus, the analysis of the current state of the art shows that despite the stated need and certain successes in applying NLP to aviation texts, a number of unresolved issues and research gaps remain. In particular, most studies on defect classification rely on a small number of failure categories, which do not fully reflect the diversity of malfunctions in real commercial aircraft. Quantitative assessments of the quality of defect classification based on maintenance text data are often lacking, which complicates the comparison of approaches and assessment of their practical applicability. Several researchers have not used English language texts in their studies, limiting the applicability of the obtained results.

There is a lack of research using in-depth thematic analysis to identify subtle failure patterns within standard categories of real-world aircraft. Existing approaches based on complex deep learning models often suffer from a lack of interpretability, which is a critical drawback in the aviation industry.

Given these gaps, this study aims to partially address them.

The purpose. The purpose of this study is to develop and test a comprehensive method for the intelligent analysis of text descriptions of defects in aviation maintenance, to improve the efficiency of fault classification, and to obtain an informative and detailed understanding of their nature using interpretable methods of machine learning and NLP.

To achieve this purpose, the following objectives are solved:

1. A process for preprocessing and normalizing a corpus of real-world text records on aircraft maintenance is being developed, including the decryption of industry-specific abbreviations and acronyms, as well as other text cleaning procedures to prepare the data for analysis.

2. An ML model is being synthesized for the automatic multi-class classification of defects into ATA Chapter-Section 100 (ATA iSpec 2200) categories based on their text descriptions.

3. An approach is being developed to identify and interpret hidden subgroups (themes) of faults within the most represented Chapter-Section (CS) category in the dataset.

Achieving this goal will enable the efficient processing of unstructured text data on maintenance, a “blind spot” for many existing systems. The practical effect will be accelerated diagnostics, reduced operating costs, and improved flight safety through a deeper, data-driven understanding of failure causes.

Materials and methods of the study. Dataset. This study utilized a dataset containing information on detected defects and maintenance actions for nine commercial aircrafts of the same model used on domestic routes between 2014 and 2020. The dataset contains 13,204 records, each of which includes the following attributes: “Date Reported” – the date of the record; “A/C Reg” – the aircraft registration number; “Defect” – a text description of the defect; “Action” – a text description of the action taken.

Additionally, the attributes include the system and subsystem codes according to the ATA 100 (ATA iSpec 2200) standard: “Chapter” – a two-digit number XX encoding the main system or area of the aircraft; “Section” – a two-digit number YY detailing the subsystem.

Fig. 1 shows a heat map illustrating the number of reported defects per year for each aircraft in the dataset. The actual aircraft registration numbers were replaced with conditional identifiers (A, B, C, etc.) to ensure confidentiality.

For ease of further analysis, the “Chapter” and “Section” features were combined into a single categorical feature, “Part”, in XX–YY format. Using this feature, 456 unique defect categories were identified in the analyzed dataset. A fragment of this dataset is listed in Table 1.

As can be seen in Table 1, the “Section” feature contains section numbers corresponding to the manufacturer’s documentation for the analyzed aircraft model, including specific codes detailing the standard ATA 100 (ATA iSpec 2200) structure.

The distribution of records across defect categories in the dataset was uneven. Fig. 2 shows the number of records for the 30 most frequently occurring defect categories, which accounted for a significant portion (approximately 62 %) of the total data.

As Table 1 illustrates, text descriptions of “Defect” defects have a specific format: they are short messages rich in industry-specific abbreviations, acronyms, and codes, typical of technical language in the aviation industry. For automated analysis, preprocessing of the available text data is required.

Preprocessing. The text data preprocessing process implemented in this study is presented in the block diagram in Fig. 3. Preprocessing was performed using specialized libraries in Python, version 3.10.

As shown in Fig. 3, at the initial processing stage, text descriptions of defects from the “Raw Text Data” dataset are analyzed for domain-specific abbreviations and acronyms for subsequent decoding. For this purpose, a special Python script was written (the “Abbreviations Detection” block in Fig. 3). The operating algorithm is as follows:

1. Tokenization of the text into words (tokens) using the regular expression “[A-Z0-9#-./]+”, which allows the correct extraction of lexemes containing numbers and special characters.

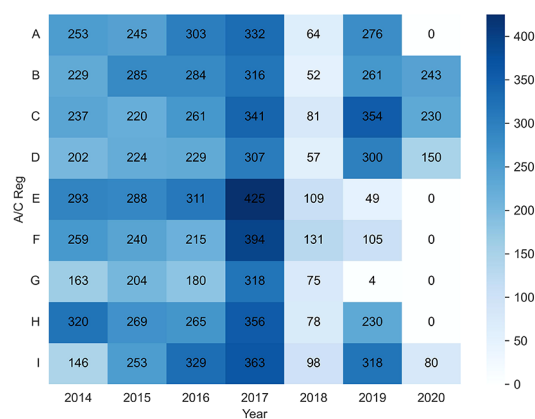


Fig. 1. Distribution of the number of defects by year for each aircraft

A fragment of the used dataset

Date Reported	A/C Reg	Part	Chapter	Section	Defect	Action
2014-01-01	B	33-10	33	10	LH DOME LIGHT IS INOP	DOME LIGHT IS REPLACED
2014-01-01	B	36-11	36	11	BLEED 2 FAIL MSG	FIM-36-10-00-810-805-A PERFORMED, NAPRSOV REPLACED IAW AMM 36-11-09. (ENG#2). ADD IS CLEARED. ADD CLEARED
2014-01-01	D	25-33	25	33	AFTER TAKE OFF BOILER AND COFFEEMAKER DIDN'T WORK IN AFT GALLEY ("WATER FAIL" FLASHED	BOILER AND COFFEEMAKER HAS BEEN ADJUSTED
2014-01-02	C	33-10	33	10	RH DOME LIGHT IS INOP	RH DOME LIGHT IS REPLACED IAW AMM 33-11-01-000- 801A. OPS TEST IS OK. ADD CLEARED
2014-01-02	F	52-10	52	10	THE DOOR 1L IS DIRTY	CABIN INTERIOR CLEANING PERFORMED

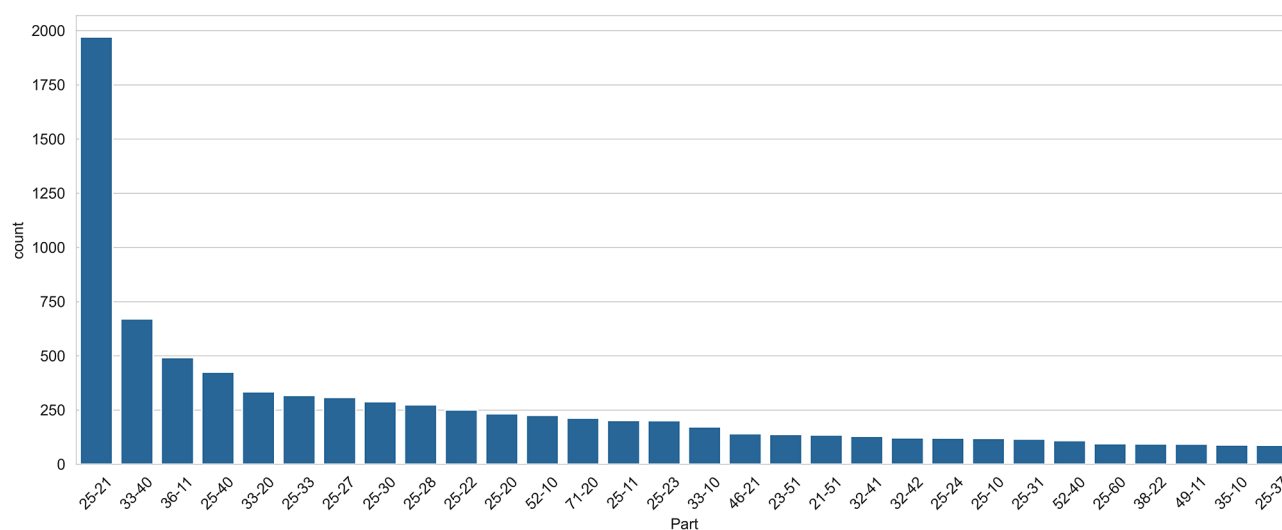


Fig. 2. The number of records for the 30 most common defect categories

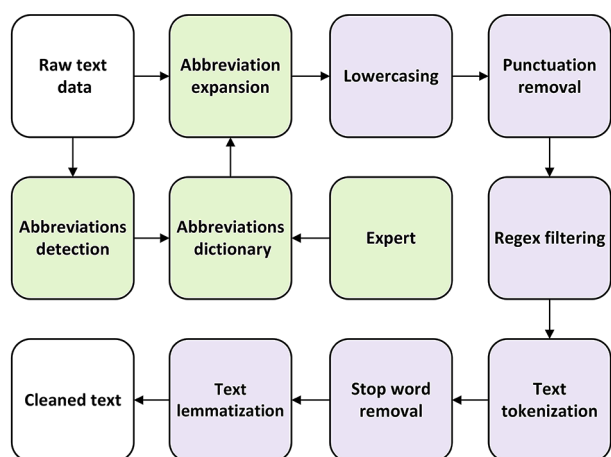


Fig. 3. Structural diagram of the text data preprocessing stage

2. A token is discarded if it does not contain letters or is included in an existing set of stop words based on the standard NLTK list [29].

3. A token is identified as a potential abbreviation if it:

- contains numbers or the symbols “.”, “/”, “#” (e.g., ENG1, MSG#2, A/C), but is not purely numeric;

- contains periods inside or at the end of a word (e.g., ENG., A.C.).

4. If a token consists solely of letters and passes the previous checks, it is compared to the standard English dictionary (NLTK words [29]). A token is also identified as a potential abbreviation if it is not listed in the dictionary and its length is within a specified limit of two to five characters. This length limit allows for a focus on the most likely abbreviations and acronyms, filtering out possible typographical errors in long words or specific technical terms.

All unique tokens identified as potential abbreviations were compiled into a single candidate list, comprising 3,549 unique lexemes.

The resulting list was then submitted to an expert from the aviation industry for further analysis to decode abbreviations and acronyms (the “Expert” block in Fig. 3).

Frequency analysis revealed that the distribution of candidate abbreviations was extremely uneven; 93.5 % of them appeared in the dataset no more than 15 times. Taking Zipf’s law into account and focusing on the most significant terms, 6.5 % of the most frequently occurring candidates were selected for manual processing by an

expert. The expert decided on the necessity and correctness of expanding each term, and excluded obvious or established terms from the dictionary that did not require replacement.

Based on the expanded data, a dictionary was created (the “Abbreviations dictionary” block in Fig. 3), which was used to replace abbreviations in the text in the “Abbreviations expansion” block.

A fragment of the resulting dictionary, indicating the number of the most frequent detections across the entire dataset, is shown in Table 2.

As shown in Table 2, some terms, such as EICAS, APU, P/N, or SEATS, are generally accepted and/or self-contained in the aviation context. Therefore, the experts did not replace them with longer equivalents to avoid technical language redundancy. Furthermore, the analysis revealed spelling and punctuation variations for the same concepts (e.g., “INOP” and “INOP”). In such cases, the expert’s task is to determine a single canonical form for the subsequent unification of these tokens during preprocessing.

It should be emphasized that the experts’ work was not entirely subjective. During the analysis, the expert relied on standardized industry sources, including glossaries defined by the ATA iSpec 2200 standard and aircraft manufacturer technical documentation. This ensures a high degree of objectivity and accuracy in the decoded terms.

As shown in Fig. 3, after replacing the abbreviations in the text, a lowercasing procedure was performed.

Because the text data for the technical documentation contain numerous serial numbers, instruction and part codes, seat codes, and other information, all numerical data were removed during the preprocessing stage to address the assigned tasks. To achieve this, punctuation marks are first removed from the technical documentation texts (the “Punctuation Removal” block in Fig. 3), with the exception of “/” and “-”, which may be part of the compound terms. Then, in the “Regex Filtering” block in Fig. 3, a multistage text filtering process is performed using regular expressions. Tokens matching code patterns (e.g., “73-31-00”), seat identifiers (e.g., “12C”), and single letters, numbers, and excess spaces are removed.

The next step involves tokenizing the test defect descriptions (the “Text tokenization” block in Fig. 3) fol-

lowed by stop word removal using tools from the NLTK 3.9.1 Python library [30]. To improve the relevance and identify more domain-specific terms, the standard stop word list was expanded. Frequently occurring but uninformative words for classification and topic modeling tasks were added, including “hand”, “inoperative”, “aft”, “forward”, “left”, “right”, “rev”, “accordance”, “ref”, “condition”, “bad”, “fail”, “message”, “doesnt”, “side”, and “broken”.

The final preprocessing step involves lemmatization (the “Text lemmatization” block in Fig. 3), also using NLTK 3.9.1 library. This procedure reduces various word forms to their basic dictionary form (lemma), which allows for the unification of text data, reduction of the dimensionality of the feature space, and improvement of the robustness of ML models.

Building a model for classifying defects by category.

The task of automatically classifying defects using standard CS codes can be used for data validation, assisting in the correct completion of defect logs and identifying potential inconsistencies. To build the classification model, we focused on the 30 most commonly represented defect categories (Fig. 2). This subsample, covering 62 % of all records, represents a compromise between the breadth of coverage and methodological rigour. This approach, on the one hand, provides a sufficient data volume for each class for reliable training and model evaluation, and on the other hand, allows for focusing the analysis on categories that have the greatest operational significance in daily maintenance practices. Rare categories were excluded from this classification task because their analysis requires the use of other approaches, such as anomaly detection methods.

In the course of experimental studies conducted using the scikit-learn 1.7.0 Python library [31], an effective multi-class classification method was developed based on feature extraction from preprocessed text data using the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm [32] followed by the application of a logistic regression model (LogReg). Because the One-vs-Rest strategy was used for multiclass classification, a separate binary model was trained for each class k out of K classes.

The LogReg classifier was optimized by selecting the following hyperparameters: *class_weight* = “balanced” – automatic weighting of classes inversely proportional to their frequencies; *C* = 2.4 – regularization parameter; *solver* = “liblinear” – optimization algorithm with l2-regularization.

The TF-IDF value for a term (word) t in document d is calculated using the formula

$$tfidf(t, d, D) = f_{t,d} \left(\log \left(\frac{1 + n_D}{1 + df(t, D)} \right) + 1 \right), \quad (1)$$

where $f_{t,d}$ is the number of occurrences of the word t in the document (text description of the defect) d ; n_D is the total number of documents in the corpus D ; $df(t, D)$ is the number of documents in the corpus D , in which the term occurs t .

The following settings were also selected for the TF-IDF algorithm: *sublinear_tf* = *False* – uses linear scaling of word frequency $TF(t, d) = f_{t,d}$ (1); *ngram_range* = (1, 2) – unigrams and bigrams are extracted;

Table 2

A fragment of the dictionary of abbreviations and their frequency in the corpus

Abbreviation	Expansion	Frequency
FWD	forward	1,087
INOP	inoperative	1,021
RH	right hand	908
MSG	message	906
EICAS	–	869
INOP.	inoperative	862
LH	left hand	847
ENG	engine	484
SEATS	–	461
APU	–	370
P/N	–	324
IAW	in accordance with	313

$min_df = 1$ – words are not ignored, even if they appear in only one document; $max_df = 0.95$ – words that appear in more than 95 % of documents are removed.

To evaluate the performance of the developed classifier, a set of standard quality metrics recommended for problems with potential class imbalance [33] was used. The $f1$ -score with macro-averaging ($f1$ -macro) was chosen as the main metric for comparing the models, since it represents the harmonic mean between precision and recall and is averaged over all classes without weighting, making it sensitive to performance on small classes.

Additionally, the following metrics were calculated:

1. *Accuracy* – total proportion of correctly classified objects.

2. *Balanced accuracy* – the arithmetic mean of (*recall*) across all classes, which allows for correct evaluation of models on unbalanced data.

3. *Precision-macro* and *recall-macro* – unweighted average precision and recall for all classes.

4. *F1-weighted* – weighted by the number of objects in each class, the average $f1$ -score, reflecting the overall performance of the classifier.

5. *Cohen's Kappa* (κ) – a multiclass version of a metric that measures the degree of agreement between model predictions and true labels, correcting for random guessing

$$\kappa = (p_o - p_e) / (1 - p_e), \quad (2)$$

where $p_o = \frac{1}{N} \sum_{k=1}^K M_{k,k}$ is proportion of observed agreement (*observed agreement*) or *accuracy* (K is the number of classes; N is the total number of all objects;

$M_{K \times K}$ is an error matrix); $p_e = \frac{1}{N^2} \sum_{k=1}^K c_k g_k$ is an expected random agreement (*expected agreement*), if the model guessed by chance (takes into account the distributions of classes in predictions and in true labels). From here

on $c_k = \sum_{j=1}^K M_{k,j}$ is the total number of objects predicted to be of class k , a $g_k = \sum_{i=1}^K M_{i,k}$ is the total number of objects that actually belong to class k .

6. The Matthews Correlation Coefficient (MCC) – correlation metric of classification quality. For a multiclass case, the MCC is calculated based on the entire confusion matrix [32]. In this paper, we use the gen-

eralized MCC formula for K classes, implemented in the scikit-learn 1.7.0 library [31]

$$MCC = \frac{N \sum_{k=1}^K M_{k,k} - \sum_{k=1}^K c_k g_k}{\sqrt{N^2 - \sum_{k=1}^K c_k^2} \cdot \sqrt{N^2 - \sum_{k=1}^K g_k^2}}. \quad (3)$$

Table 3 presents the performance metrics for multiclass defect classification using standard CS codes, obtained for the proposed TF-IDF & LogReg methods using five-fold stratified cross-validation.

Furthermore, Table 3 provides a comparison of the results for the Bag-of-Words algorithm (CountVectorizer in the scikit-learn library) [34], which demonstrated the best performance when paired with a linear support vector machine classifier (CountVectorizer & LinearSVC). Table 3 also presents the performance metrics for a method based on word vector representations obtained using the Word2Vec model [35] and subsequent classification using logistic regression (Word2Vec & LogReg).

Additionally, a solution (Fine-tuning BERT in Table 3) using retraining of a deep artificial neural network with a transformer architecture from the transformer library [36] was investigated. For this purpose, the pre-trained DistilBERT (distilbert-base-uncased) model was chosen, which is an efficient and widely used version of the BERT model [37].

As shown in Table 3, the proposed TF-IDF & LogReg method demonstrated the best average values for most key metrics, outperforming more complex approaches, including retraining the transformer model. However, a paired t -test revealed no statistically significant superiority of the TF-IDF & LogReg model over the CountVectorizer and LinearSVC model ($p > 0.05$). Therefore, it can be concluded that both models demonstrate comparable high performance for this task.

Despite the lack of statistically significant differences, the TF-IDF and LogReg combination was selected as the primary model for further in-depth analysis in this study. This choice was based on two key advantages critical for applied problems in the aviation industry: the high interpretability and probabilistic nature of the model.

Fig. 4 shows a heatmap for the developed TF-IDF and LogReg classification method, reflecting the *precision*, *recall*, and *f1-score* metrics for the 30 selected classes. As *support (norm)*, Fig. 6 shows the value of the

Table 3

Comparison of classifiers using five-fold stratified cross-validation

Metrics	TF-IDF & LogReg		CountVectorizer & LinearSVC		Word2Vec & LogReg		Fine-tuning BERT	
	mean	std	mean	std	mean	std	mean	std
<i>Accuracy</i>	0.8255	0.0015	0.8251	0.0026	0.7977	0.0063	0.8258	0.0044
<i>Balanced accuracy</i>	0.7652	0.0044	0.7617	0.0070	0.7365	0.0063	0.7075	0.0098
<i>Precision macro</i>	0.7674	0.0080	0.7648	0.0079	0.7244	0.0090	0.7079	0.0241
<i>Recall macro</i>	0.7652	0.0044	0.7617	0.0070	0.7365	0.0063	0.7075	0.0098
<i>F1-macro</i>	0.7619	0.0062	0.7589	0.0077	0.7261	0.0068	0.6912	0.0090
<i>F1-weighted</i>	0.8219	0.0023	0.8211	0.0030	0.7953	0.0061	0.7997	0.0042
$k(2)$	0.8091	0.0016	0.8086	0.0031	0.7793	0.0068	0.8085	0.0048
$MCC(3)$	0.8093	0.0016	0.8088	0.0031	0.7795	0.0068	0.8094	0.0048

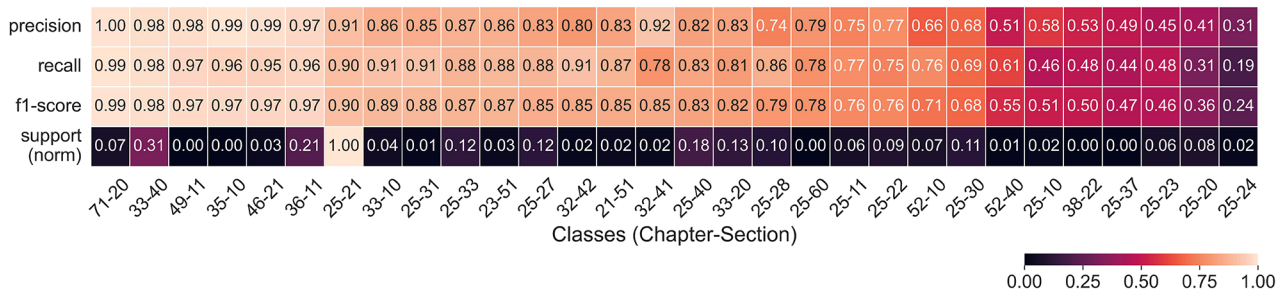


Fig. 4. Precision, recall, and f1-score metric values by class for the proposed TF-IDF & LogReg classification method

number of objects in the class normalized to the interval [0, 1].

To evaluate the interpretability of the developed classifier model and identify the most significant features for each defect category, the weighting coefficients of the trained logistic regression models were analyzed. The probability of belonging to a defect description i (represented by the TF-IDF feature vector x_i) of class k is determined by a linear combination of the features

$$a_k(x) = w_{k,0} + \sum_{j=1}^p w_{k,j} x_{i,j}, \quad (4)$$

where $x_{i,j}$ is the TF-IDF value of the j^{th} token (word) in the i^{th} document and $w_{k,j}$ is the corresponding weight coefficient of the trained model for class k .

The significance of the j^{th} token (word) for determining membership in class k is directly proportional to the magnitude of its weight coefficient $w_{k,j}$. High positive values of $w_{k,j}$ indicate that this token is a strong indicator of class k .

Table 4 lists the seven tokens with the highest positive weights for the four defect categories.

The analysis in Table 4 shows that the developed classifier makes decisions based on logical and interpretable features.

Search and interpretation of hidden subgroups of faults. The ATA 100 (ATA iSpec 2200) coding system used in aviation maintenance may not be detailed enough to understand the specifics of emerging faults and identify their root causes within a single CS group [27]. This limitation complicates the development of

targeted preventive measures and the optimization of maintenance programs.

This study proposes an approach for conducting a more in-depth analysis of defects grouped within a single CS using a topic modeling method based on latent Dirichlet allocation (LDA).

LDA is a probabilistic generative model based on the Bayesian approach [38]. LDA considers each document d from a corpus D as a mixture of latent topics, and each topic k from a total set of K topics as a probability distribution over words from vocabulary V .

The topic modeling task is reduced to calculating the posterior distribution $p(Z, \Theta, \Phi | T, \alpha, \beta)$, where T is the observed words, and Z, Θ, Φ are latent variables (word topics, topic distributions in documents, and word distributions in topics, respectively). Because the exact calculation of this distribution is an intractable problem, an iterative variational Bayesian inference algorithm implemented in the scikit-learn 1.7.0 library [31] was used to approximate it.

One of the most numerous defect categories with the “Part” feature, 25–21, containing 1971 records, was chosen as the object of topic modeling. This category corresponds to the CS “25–20 EQUIPMENT/FURNISHINGS-Passenger Compartment” according to ATA 100 (ATA iSpec 2200) code. The proposed analysis methodology is as follows: After preprocessing, all text descriptions of defects with a “Part” attribute equal to 25–21 were filtered from the overall dataset according to the diagram in Fig. 4.

Because LDA operates on word frequencies, Count-Vectorizer from the scikit-learn library was used to

Table 4

The most significant tokens for four defect categories, obtained from the weights of the trained logistic regression model

ATA 100 code (ATA iSpec 2200)	25-20 EQUIPMENT/FURNISHINGS-Passenger Compartment		33-40 LIGHTS-Exterior		36-10 PNEUMATIC-Distribution		52-10 DOORS-Passenger/Crew	
The “Part” feature in the dataset	25–21		33–40		36–11		52–10	
	Token	Weight	Token	Weight	Token	Weight	Token	Weight
1	seat	8.2114	light	9.6961	bleed	15.6990	door	12.6649
2	seatback	5.0030	taxi	5.8214	psi	4.1701	pax door	4.8046
3	backrest	3.6838	navigation	5.3608	qrh	3.6731	cockpit door	4.1663
4	table	3.6816	wing	4.1136	naprsov	3.4183	cockpit	3.5356
5	pocket	3.1865	navigation light	4.0800	eicas	3.0961	pax	3.0052
6	recline	3.0330	taxi light	3.5350	fluctuation	2.8052	lock	2.9383
7	back	2.5706	nose	3.4822	valve	2.4427	noise	2.8998

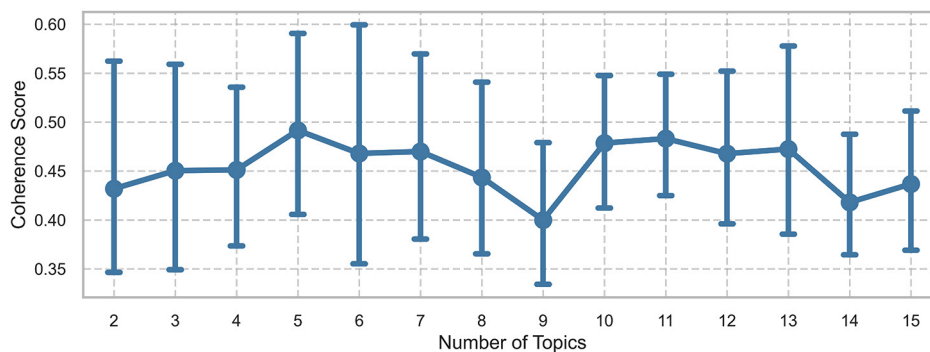


Fig. 5. Dependence of the average coherence C_v on the number of topics

transform the text corpus into a document-term matrix. To reduce noise and filter out uninformative words, the vectorizer was configured with the following parameters: $min_df=2$, $max_df=0.95$.

The LatentDirichletAllocation implementation from the Scikit-learn library was used to build the topic model. The key hyperparameter of the LDA model, which determines the number of topics ($n_components$), was obtained through a quantitative analysis aimed at finding an optimal and stable solution. A series of computational experiments was conducted to train LDA models with the number of topics k ranging from 2 to 15. To assess the robustness of the thematic structure, the coherence metric C_v was repeatedly calculated for each value of k while varying the number of most probable words (top_words) from 2 to 10. The results of this experiment, averaged over all top_words values for each k with a 95 % confidence interval, are shown in Fig. 5.

The graph in Fig. 5 shows a peak in the average coherence at $k=5$. In addition to the quantitative analysis, an expert assessment of the themes' interpretability was performed. The model with $k=5$ allowed us to identify semantically coherent and practically meaningful fault groups. Models with a larger number of themes ($k > 8$), although having narrower confidence intervals (Fig. 5), demonstrate excessive fragmentation of existing themes into less-meaningful subgroups.

The resulting themes were analyzed based on the lists of the most probable words for each theme, as shown in Fig. 6.

Results discussion. To address the central research objective of automatically classifying defects into 30 CS categories, four different approaches were tested and compared. The results of the five-fold stratified cross-validation for all the models are presented in Table 3.

The analysis showed that the two methods demonstrated the best performance. The method based on TF-IDF and logistic regression (TF-IDF & LogReg) demonstrated an $f1_macro$ value of 0.7619. The approach based on bag-of-words and a support vector machine (CountVectorizer & LinearSVC) demonstrated an $f1_macro$ value of 0.7589.

This confirms the hypothesis that the presence or absence of keywords is a strong signal for classifying short technical texts.

Approaches using semantic vector representations demonstrated lower performances. The Word2Vec & LogReg model achieved an $f1_macro$ of 0.7261, which may be due to the insufficient data in the highly specialized corpus for training high-quality embeddings. Most telling is the result of retraining the DistilBERT transformer model (a fine-tuning BERT method): contrary to expectations for a state-of-the-art architecture, it demonstrated the lowest performance ($f1_macro = 0.6912$). This significant result is likely due to the excessive complexity of the contextual analysis for this task and the insufficient data volume for effective model adaptation to the aviation domain. This observation highlights that for certain applied problems, classical but well-tuned and interpretable methods may be more effective.

Given the good interpretability and probabilistic nature of logistic regression, the TF-IDF & LogReg methods were subjected to a more detailed analysis. This classifier model demonstrated high results according to the used metrics: $balanced_accuracy = 0.7652$, $precision_macro = 0.7674$, $recall_macro = 0.7652$, $f1_weighted = 0.8219$, Cohen's kappa $\kappa = 0.8091$. The obtained metrics indicate that the model works effectively, even with class imbalance. The high performance of the model was also confirmed by the analysis of Cohen's kappa (k) and

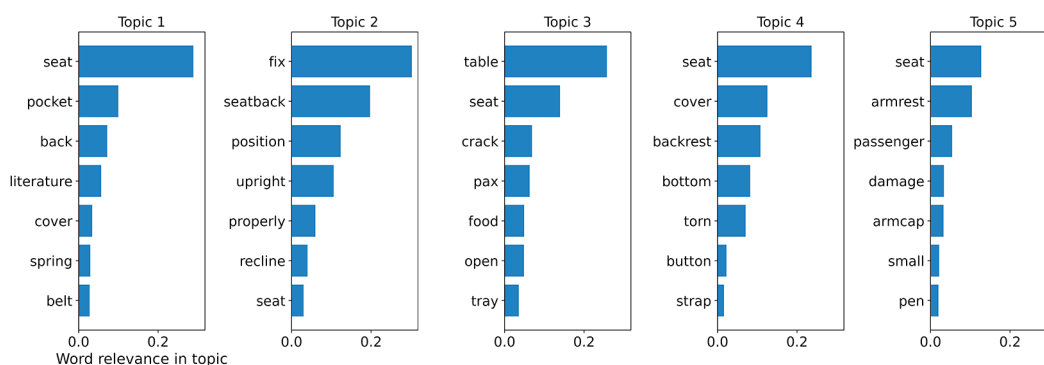


Fig. 6. The results of topic modeling in the form of lists of the most probable words for each topic

MCC metrics. Both metrics evaluate the quality of the classification corrected for random guessing, making them especially valuable for problems with imbalanced classes. For the TF-IDF & LogReg classifiers, the *MCC* value was 0.8093. It is known that κ and *MCC* values in the range of 0.8–1.0 are traditionally interpreted as “very good,” meaning that the model’s performance significantly exceeds that of a random classifier. This indicates that the model has extracted meaningful patterns from the text and has not overfitted the data. The close values obtained for Cohen’s kappa κ and *MCC* were not anomalies. As shown in a previous study [39], especially with some class imbalance, these two metrics often yield very similar model performance estimates.

The low standard deviation of the metrics (0.0015–0.008) for TF-IDF & LogReg indicates the high stability and robustness of the model.

A more detailed analysis of the TF-IDF & LogReg model’s performance for each class (Fig. 4) allows us to draw several important conclusions about the nature of the data and the capabilities of the proposed approach.

First, the analysis confirmed the absence of a direct linear relationship between the sample size and classification quality. For example, the CS 71–20 class (213 examples) demonstrated near-perfect performance (*f1-score* = 0.99), outperforming the largest class, CS 25–21 (1971 examples, *f1-score* = 0.90). This can be explained by lexical uniqueness: categories associated with specific technical systems (engine, pneumatics) use a specific vocabulary that allows the model to uniquely identify them. Simultaneously, the semantically heterogeneous category CS 25–21, as shown by topic modeling, complicates the classification task despite the large data volume.

Secondly, the analysis reveals a number of “problematic” classes (e.g., CS 25–24, 25–20, 25–23) with low *f1-scores* (0.24–0.50). Their low classification quality is likely due to a combination of factors: lexical overlap with larger classes (especially within Chapter 25), lack of unique terminology, and an insufficient number of training examples against a backdrop of high semantic uncertainty.

Third, analyzing metrics by class allows us to evaluate the model performance in the context of practical applications, where not all errors are created equally. In aviation maintenance, missing a real defect (False Negative) in a critical system can have more serious consequences than a false alarm (False Positive). In this regard, special attention should be paid to the *recall* metric, which reflects the completeness of the defect detection.

For most technically significant systems, the model demonstrated high *recall*: 71–20 (Power Plant) – 0.99, 36–11 (Pneumatic) – 0.96, 33–40 (Lights Exterior) – 0.98, which is a strong positive result from a safety perspective. However, for the other classes, there was a significant imbalance between the *precision* and *recall*. For example, for class CS 52–10 (Doors), a higher *recall* (0.76) with low *precision* (0.66) indicates that the model is good at detecting door defects but is prone to false alarms. Conversely, for CS 32–41 (Landing Gear), a high *precision* (0.92) with a lower *recall* (0.78) indicates that the model’s predictions are very reliable, but it risks missing approximately 22 % of real-world defects in this system, which requires attention during practical implementation.

Notably, in the overall model comparison (Table 3), the best approaches (TF-IDF & LogReg and Count-Vectorizer & LinearSVC) outperform the alternatives not only in terms of *f1-macro* but also in terms of *recall macro*, confirming their overall high fault detection capability.

These findings also demonstrate the importance of data volume in supervised learning tasks. Increasing the training set size by collecting data over a longer period or incorporating data from other computers is the most direct and effective way to improve the classification reliability for these “challenging” classes. A larger dataset will not only allow the model to better understand the specifics of existing minority categories but will also pave the way for including rarer defect types in the analysis that were excluded in this stage of the study.

Overall, the obtained results demonstrate that the proposed TF-IDF & LogReg classification method is highly effective for most technically specific and well-represented defect categories.

Furthermore, an analysis of the most significant tokens for various classes (Table 4) confirmed that the LogReg model was trained on logical and easily interpretable features. For example, for CS 25–21 (EQUIPMENT/FURNISHINGS – Passenger Compartment), the words “seat,” “seatback,” “backrest,” and “table” have the highest weight, while for CS 36–11 (PNEUMATIC – Distribution), the words “bleed,” “psi,” “qrh,” and “naprsov” have the highest weight. This confirms the accuracy of the model and emphasizes its interpretability, which is critical for the aviation industry.

To demonstrate the capabilities of the in-depth analysis, a topic modeling experiment using LDA was conducted for the most numerous defect category, CS 25–21. The results (Figs. 5 and 6) convincingly demonstrate that the model successfully decomposed this general category into five clearly defined and easily interpretable thematic clusters.

Expert keyword analysis for each topic yielded the following interpretations:

1. *Topic 1*: “Pocket and belt defects” (words: seat, pocket, back, literature, cover, spring, belt).
2. *Topic 2*: “Seat mechanism malfunctions” (words: fix, seatback, position, upright, properly, recline, seat).
3. *Topic 3*: “Damage to folding tables” (words: table, seat, crack, pax, food, open, tray).
4. *Topic 4*: “Damage to upholstery and seat covers” (words: seat, cover, backrest, bottom, torn, button, strap).
5. *Topic 5*: “Armrest Damage and Minor Defects” (words: seat, armrest, passenger, damage, armcap, small, pen).

This result has significant practical implications. Instead of a single general problem according to CS 25–21, reliability engineers and maintenance planners receive a detailed picture, enabling targeted root cause analysis, optimization of specific spare parts inventory, and planning of work for different specialists (mechanics, interior specialists). Thus, topic modeling serves as a powerful tool for a more detailed analysis than that possible with standard CS codes alone.

Conclusions. In this study, we developed and validated a comprehensive approach for analyzing unstructured text descriptions of defects from real-world air-

craft maintenance data. This study aimed to address the challenges of automatically classifying defects and obtaining a detailed understanding of their nature using topic modeling.

The key result of this study is the creation and validation of a performant, interpretable classification model. A comparative analysis showed that the classical methods were the most effective for this task. In particular, the approach based on TF-IDF and logistic regression achieved high-quality metrics, demonstrating a performance that is statistically comparable to the strong competitors CountVectorizer and LinearSVC. Owing to its interpretability and probabilistic nature, a logistic regression-based model was selected for an in-depth analysis. Both of these classical methods significantly outperformed the approaches using semantic embeddings (Word2Vec) and transformer model retraining (DistilBERT). The success of the classifiers is largely due to the proposed preprocessing process, which includes expert deciphering of the domain abbreviations.

Furthermore, it was demonstrated that applying topic modeling (LDA) to one of the major defect categories allows for its successful decomposition into five specific and easily interpretable subgroups. This confirms the ability to extract detailed knowledge inaccessible through analysis at the level of standard maintenance codes and demonstrates that this method can be extended to other categories of maintenance codes.

Despite its demonstrated effectiveness, the proposed approach has limitations related to the specificity of the dataset, sample size for minority classes, and labor-intensive preprocessing stage. Further research should aim to validate the methodology on larger datasets, explore hierarchical approaches for complex classes, and develop semi-automated methods for creating domain dictionaries. Overall, this study contributes to the development of a methodology for analyzing unstructured technical texts and demonstrates how the application of a balanced, interpretable approach allows for the effective extraction of valuable knowledge from aviation maintenance data, contributing to improved flight safety and optimization of operational processes.

References.

1. Dhillon, B. S., & Liu, Y. (2006). Human error in maintenance: a review. *Journal of Quality in Maintenance Engineering*, 12(1), 21-36. <https://doi.org/10.1108/13552510610654510>
2. Agustian, E. S., & Pratama, Z. A. (2024). Artificial Intelligence Application on Aircraft Maintenance: A Systematic Literature Review. *EAI Endorsed Trans IoT*, 10.
3. Kalantayevskaya, N., Koshekov, K., Latypov, S., Savostin, A., & Kunelbayev, M. (2022). Design of decision-making support system in power grid dispatch control based on the forecasting of energy consumption. *Cogent Engineering*, 9(1). <https://doi.org/10.1080/23311916.2022.2026554>
4. Sathyananda Swamy, H. V., Manoj, B. N., Zaiba, N., & Pandey, M. (2024). *A Study of Artificial Intelligence in Aviation Management*, (pp. 108-114). QTanalytics Publication (Books). <https://doi.org/10.48001/978-81-966500-8-7-11>
5. Errico, A., Travascio, L., & Vozella, A. (2025). Analysis of Safety Metrics Supporting Air Traffic Management Risk Models. *Engineering Proceedings*, 90, 43. <https://doi.org/10.3390/engproc2025090043>
6. Jammal, P., Pinon-Fischer, O., Mavris, D., & Wagner, G. (2025). Predictive Maintenance of Aircraft Braking Systems: A Machine Learning Approach to Clustering Brake Wear Patterns. *AIAA SciTech 2025 Forum*. <https://doi.org/10.2514/6.2025-0710>
7. Savostin, A., Koshekov, K., Tuleshov, A., Savostina, G., & Koshekov, A. (2024). Development of remote diagnostic monitoring system for pumping equipment with open architecture. *Radioelectronic and Computer Systems*, 4(112), 192-206. <https://doi.org/10.32620/reks.2024.4.16>
8. Sundaram, S., & Zeid, A. (2025). Technical language processing for Prognostics and Health Management: applying text similarity and topic modeling to maintenance work orders. *Journal of Intelligent Manufacturing*, 36, 1637-1657. <https://doi.org/10.1007/s10845-024-02323-4>
9. Kretz, D. R. (2018). Experimentally Evaluating Bias-Reducing Visual Analytics Techniques in Intelligence Analysis. In Ellis, G. (Ed.). *Cognitive Biases in Visualizations*. Cham, Springer. https://doi.org/10.1007/978-3-319-95831-6_9
10. Nanyonga, A., Joiner, K., Turhan, U., & Wild, G. (2025). Applications of Natural Language Processing in Aviation Safety: A Review and Qualitative Analysis. *Reliability Engineering & System Safety*, (in press). <https://doi.org/10.48550/arXiv.2501.06210>
11. Rogers, F., Kovaleva, O., & Rumshisky, A. (2020). A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, 8, 842-866. https://doi.org/10.1162/tacl_a_00349
12. Nanyonga, A., Wasswa, H., Joiner, K., Turhan, U., & Wild, G. (2025). Explainable Supervised Learning Models for Aviation Predictions in Australia. *Aerospace*, 12, 223. <https://doi.org/10.3390/aerospace1203022>
13. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., ..., & Liu, T. (2024). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM TOIS*, (in press). <https://doi.org/10.48550/arXiv.2311.05232>
14. Kosch, T., & Feger, S. (2024). Risk or Chance? Large Language Models and Reproducibility in HCI Research. *ACM Interactions*. <https://doi.org/10.48550/arXiv.2404.15782>
15. Alomar, I., & Nikita, D. (2025). Managing Operational Efficiency and Reducing Aircraft Downtime by Optimization of Aircraft On-Ground (AOG) Processes for Air Operator. *Applied Sciences*, 15, 5129. <https://doi.org/10.3390/app15095129>
16. NASA. *Aviation Safety Reporting System*. Retrieved from <https://asrs.arc.nasa.gov/>
17. Rose, R. L., Puranik, T. G., & Mavris, D. N. (2020). Natural Language Processing Based Method for Clustering and Analysis of Aviation Safety Narratives. *Aerospace*, 7, 143. <https://doi.org/10.3390/aerospace7100143>
18. Kuhn, K. D. (2018). Using structural topic modeling to identify latent topics and trends in aviation incident reports. *Transportation Research Part C: Emerging Technologies*, 87, 105-122. <https://doi.org/10.1016/j.trc.2017.12.018>
19. *National Transportation Safety Board*. Retrieved from <https://www.ntsb.gov/>
20. Kierszbaum, S., & Lapasset, L. (2020). Applying Distilled BERT for Question Answering on ASRS Reports. *2020 New Trends in Civil Aviation (NTCA)*, (pp. 33-38). Prague, Czech Republic. <https://doi.org/10.23919/NTCA50409.2020.9291241>
21. Dong, T., Yang, Q., Ebadi, N., Luo, X. R., & Rad, P. (2021). Identifying Incident Causal Factors to Improve Aviation Transportation Safety: Proposing a Deep Learning Approach. *Journal of Advanced Transportation*, 2021, 1-15. <https://doi.org/10.1155/2021/5540046>
22. Nanyonga, A., Joiner, K., Turhan, U., & Wild, G. (2025). Applications of Natural Language Processing in Aviation Safety: A Review and Qualitative Analysis. *AIAA 2025-2153 Session: AI/ML and Autonomy Software Engineering Practices*. <https://doi.org/10.2514/6.2025-2153>
23. Wang, L., Chou, J., Rouck, D., Tien, A., & Baumgartner, D. M. (2023). *Adapting Sentence Transformers for the Aviation Domain*. <https://doi.org/10.48550/arXiv.2305.09556>
24. Akhbardeh, F., Desell, T., & Zampieri, M. (2020). MaintNet: A collaborative open-source library for predictive maintenance language resources. M. Ptaszynski & B. Ziolkowski (Eds.). *Proceedings of the 28th international conference on computational linguistics: System demonstrations. International Committee on Computational Linguistics (ICCL)*, (pp. 7-11). <https://doi.org/10.18653/v1/2020.coling-demos.2>
25. Air Transport Association of America (2021). *iSpec 2200: Information Standards for Aviation Maintenance*. Harvard Dataverse. <https://doi.org/10.7910/DVN/G1DSMX>
26. Zhou, S., Chen, B., Zhang, Y., Liu, H., Xiao, Y., & Pan, X. (2020). A Feature Extraction Method Based on Feature Fusion and its Application in the Text-Driven Failure Diagnosis Field. *International Journal of Interactive Multimedia and Artificial Intelligence*, 6, 121-130. <https://doi.org/10.9781/ijimai.2020.11.006>
27. Xu, Z., Chen, B., Zhou, S., Chang, W., Ji, X., Wei, C., & Hou, W. (2021). A Text-Driven Aircraft Fault Diagnosis Model Based on a

- Word2vec and Prior-Knowledge Convolutional Neural Network. *Aerospace*, 8, 112. <https://doi.org/10.3390/aerospace8040112>
28. Scott, M. J. (2024). Application of natural language processing for aircraft defect tracking in maintenance operations. *ICAS PROCEEDINGS 34th Congress of the International Council of the Aeronautical Sciences*, Florence, Italy. 2024.
29. *Natural Language Toolkit (NLTK). NLTK Data – Words Corpus* (n.d.). Retrieved from https://www.nltk.org/nltk_data
30. *Natural Language Toolkit (NLTK)* (n.d.). Retrieved from <https://www.nltk.org>
31. Scikit-learn developers. *Scikit-learn: Machine Learning in Python, version 1.7.0*. (n.d.). Retrieved from <https://scikit-learn.org/stable/>
32. Gorodkin, J. (2004). Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, 28, 367-374. <https://doi.org/10.1016/j.compbiolchem.2004.09.006>
33. Powers, D. (2011). Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies*, 2, 37-63.
34. Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
35. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. arXiv, arXiv:1301.3781.
36. Wolf, T., Debut, L., Sanh, V., & Chaumond, J. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. ACL*, 38-45.
37. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. arXiv, 2019. arXiv:1910.01108.
38. Blei, D. M., Ng, A. Y., & Jordan, M.I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, 993-1022.
39. Chicco, D., Warrens, M. J., & Jurman, G. (2021). The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment. *IEEE Access*, 9, 78368-78381. <https://doi.org/10.1109/ACCESS.2021.3084050>

Комплексний аналіз текстових звітів про авіаційне технічне обслуговування із використанням методів обробки природної мови

О. Савостін¹, Г. Каїпбек^{*2}, К. Кошеков²,
Г. Савостіна¹, К. Уордл^{2,3}

1 – Північно-казахстанський університет імені М. Козібаєва, м. Петропавлівськ, Республіка Казахстан

2 – Академія цивільної авіації, м. Алмати, Республіка Казахстан

3 – АТ «Ейр Астана», м. Алмати, Республіка Казахстан

* Автор-кореспондент e-mail: kaipbekgulsanat@gmail.com

Мета. Дослідження спрямоване на розроблення й апробацію комплексного методу аналізу неструктурованих текстових описів дефектів, отриманих із реальних даних авіаційного технічного обслуговування (ТО). Метою є підвищення ефективності та глибини аналізу несправностей шляхом розв'язання двох ключових завдань: автоматичної класифікації дефектів за стандартними категоріями й ви-

явлення прихованих тематичних підгруп у межах цих категорій.

Методика. У роботі використано набір даних записів про ТО дев'яти комерційних повітряних суден за семирічний період. Розроблено багатоетапний процес попередньої обробки, що включає алгоритм ідентифікації й експертного розшифрування галузевих скорочень. Для розв'язання задачі багатокласової класифікації за 30 категоріями Chapter-Section проведена систематична оцінка та порівняння різних моделей машинного навчання. У роботі наведені результати для чотирьох репрезентативних підходів: CountVectorizer із LinearSVC, TF-IDF і Word2Vec із логістичною регресією, а також донавчання трансформерної моделі DistilBERT. Для поглибленого аналізу найчисленнішої категорії дефектів застосоване тематичне моделювання на основі латентного розподілу Діріхле з кількісним добром оптимальної кількості тем.

Результати. Найкращі показники класифікації продемонстрував підхід на основі TF-IDF і логістичної регресії, досягнувши F1-масо = 0,762 і коефіцієнта Каппа Коена = 0,809, що статистично порівняно з CountVectorizer і LinearSVC. Класичні підходи перевершили нейромережеві моделі, що свідчить про їхню стійкість до аналізу коротких технічних текстів. Тематичне моделювання успішно декомпозувало домінуючу категорію дефектів на п'ять легко інтерпретованих і семантично цілісних підгруп.

Наукова новизна. Полягає в розробленні та апробації формалізованого методу аналізу неструктурованих даних авіаційного ТО, реалізованого як єдиний інтегрований процес. У роботі також представлено детальний порівняльний аналіз продуктивності класичних і сучасних NLP-моделей на спеціалізованих даних авіаційного технічного обслуговування.

Практична значимість. Робота має прикладний характер і містить результати, готові до впровадження. Створено прототип автоматизованого класифікатора, здатного обробляти основний потік щоденних звітів про дефекти, що зменшує час їх ручного опрацювання. Також розроблено інструмент поглибленого аналізу відмов, що забезпечує перехід від загальних кодів несправностей до аналізу конкретних підпроблем. Це сприяє оптимізації програм ТО, удосконаленню процедур діагностики й, зрештою, підвищенню безпеки польотів.

Ключові слова: обробка природної мови, технічне обслуговування, повітряне судно, класифікація, тематичне моделювання

The manuscript was submitted 03.09.25.